Homecomputer

3 1. Jahrgang

März '83 **5,50 DM** 45 öS 6,00 sfr

Neu! monatlich bei Ihrem Zeitschriftenhändler

In diesem Heft:

VC-20
Zeichengenerator
Disassembler
Slalom
Robot

ZX-81
Mint – der Geldspielautomat
U-Boot-Jagd

VC-64
Labyrinth
Roulette
Disassembler
Biorhythmus

TI-99/4A
Autokosten
Dive Bomber

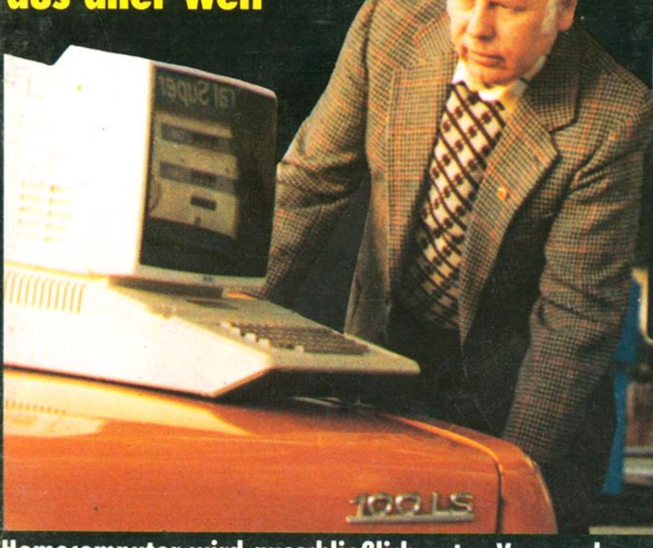
Apple II
Disk-Handler
Bundesliga

Sharp MZ 80
Submarine-Hunt

Jeden Monat mehr als 15 neue Programme + Tips + News aus aller Welt

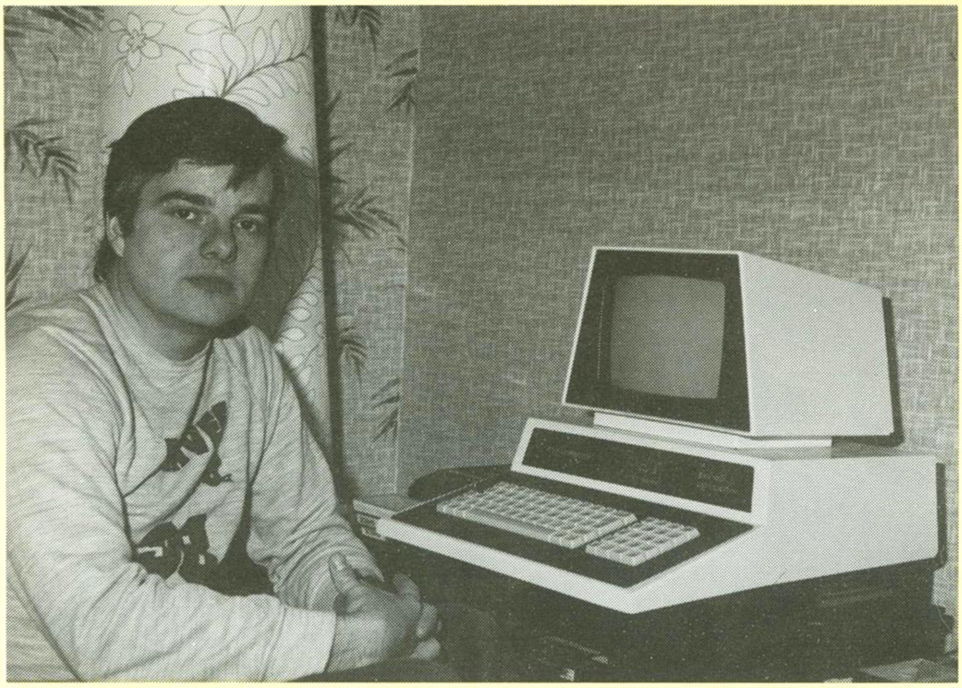






Große Werbeaktion! 250 Computer zu gewinnen!

Homecomputer wird ausschließlich unter Verwendung von Microcomputern erstellt



Liebe Leserin, lieber Leser,

Sie halten die erste, über den Handel erhältliche Zeitschrift der Welt in Händen, die ausschließlich mit Microcomputern erstellt wurde. Jede Druckzeile, Textverarbeitungsprogramm für Apple II, Druckprogramm für den Matrixdrucker EPSON MX 82, alles, was dieses Heft enthält und alles womit sein Inhalt erstellt wurde, ist in unserem Hause in eineinhalbjähriger Vorarbeit entstanden.
Zwar konnte nicht alles, was wir noch gerne im Heft abgedruckt hätten, verwirklicht werden. Erst im Laufe der nächsten Wochen werden wir unseren Mitarbeiterstab vergrößern, und erst dadurch werden wir in der Lage sein, Programme für weitere Microcomputer in Homecomputer abzudrucken; ich denke da vor allem an Atari, Tandy, Dragon, usw.
Wir tun unser Bestes, ein originelles, sich von allen anderen Computerzeischriften unterscheidendes Magazin herzustellen und haben uns bemüht, interessante Programme aufzunehmen. Sie können sich vorstellen, wie gespannt wir auf Ihre Reaktion warten. Schreiben Sie uns Ihre Meinung, denn nur so können wir sehen, was zu verbessern ist; was die Mehrzahl der Homecomputeranwender von einem Homecomputermagazin erwartet.
Jede Zeitschrift hat ihre Richtung - keine kann alles bieten. Wir können und wollen nicht in ausführlicher Form berichten, wer, wann, wo, wieviel Geld durch die Anschaffung eines Computers gespart hat. Wir wollen auch keine philosophischen Betrachtungen anstellen, wie die Welt der Computer in hundert Jahren einmal aussehen könnte. Uns genügt es, wenn Sie heute 2, 3 oder mehr Programme für Ihren Homecomputer erhalten, dies zum Preis eines Magazins und damit Ihren Spaß haben; wenn Sie nun Ihre Programme leichter erstellen können, weil Sie das eine oder andere Programm, das Sie sonst eine Menge Geld gekostet hätte, zwar in stark vereinfachter Form, somit aber auf Ihren tatsächlichen Anwendungsbereich zugeschnitten, erhielten.
Daß auch die Preise für Kleinanzeigen (Kontakte, Gebrauchtcomputermarkt, Soft- und Hardwarehandel) auf den Heimcomputermarkt zugeschnitten sind, versteht sich beinahe von selbst.
Zum Schluß möchte ich Sie noch einmal an unsere Werbeaktion erinnern. In den ersten Wochen ist es natürlich leichter, die Führung und somit die Anwartschaft auf Ihren Traum-Computer zu gewinnen. Rechnen Sie die Chancen durch; es lohnt sich sicher.

Herzlichst
Ihr Ralph Roeske

Herausgeber, Chefredakteur

Inhalt

Impressum

Homecomputer erscheint monatlich im:
Roeske Verlag, Eschwege

HERAUSGEBER:
Ralph Roeske

REDAKTION:
Ralph Roeske (R),
Chefredakteur
(verantwortlich)
Horst Brand (Br)
Georg Schröder (Sch)

HERSTELLUNG:
Roeske-Verlag
Eschwege

REPRODUKTION:
Intal Composing GmbH
3436 Hessisch Lichtenau

DRUCK:
Vogt GmbH
Druckerei + Verlag
3436 Hessisch Lichtenau

VERTRIEB:
Inland (Groß-, Einzel- und Bahnhofsbuchhandel), sowie Österreich und Schweiz:
Verlagsunion
Friedrich-Bergius-Str. 7
6200 Wiesbaden
Tel.: 06121-2660

ANZEIGENLEITUNG:
Monika Roeske (M)

ERSCHEINUNGSWEISE:
Erstverkaufstag von Homecomputer ist jeweils der letzte Montag vor dem Erscheinungsmonat

ANSCHRIFT:
Homecomputer
Gartenstraße 47
3440 Eschwege
Tel.: 05651-5993

BEZUGSPREISE:
Einzelheft: 5,50 DM
Abonnement: Inland 55,- DM pro Jahr (12 Ausgaben)
Ausland: Europa 80,-DM
USA 110,-DM

ANZEIGENPREISE:
Es gilt die Anzeigenpreisliste Nr.1 vom 1.Januar 1983
1/1 Seite S/W 3000,-DM
Farbzuschlag je Farbe (Europaskala) 500,-DM

AUTOREN, MANUSKRIPTE:
Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen. Honorare nach Vereinbarung. Bei Zusendung von Manuskripten und Software erteilt der Autor dem Verlag die Genehmigung zum Abdruck.
Rücksendung erfolgt nur bei angeforderten Beiträgen, ansonsten nur gegen Erstattung der Unkosten.
Zusendungen von Software zur Veröffentlichung sollte bitte folgendes enthalten: Kopierfähige Kassette oder Diskette mit dem Programm, von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinen-Listings), eventuell Bildschirmfotos von einem Probelauf.

Im Gespräch

Der Apple lernt sehen 4
Der Video-Interface-Spezialist
Lothar Bartscher und seine Werke

Produkte

Neu auf dem Mikromarkt 7

News

Neues aus der Branche 8

64'VideoChips

Biorhythmus-Programm 10
Roulette 12
Labyrinth 15

PET-Bytes

Concorde 18
Kalender & Uhr 18
51 - das Kartenspiel 20

Sharp MZ 80

Submarine-Hunt 22

Die ZX-Seiten

Der kleine Schwarze... 25
Der ZX-Spectrum unter der Lupe
ZX-U-Boot-Jagd 26
Rotamint 27
der Geldspielautomat

Apple-Kiste

Apple II Diskhandler 30
Apple-Steno Groß/Klein 32

Starship VC-20

VC-Z61/HC 42
der Zeichengenerator
Disassembler 47
Slalom 50
Robot 52

OPEN4,4: LIST

Bundesliga 54
Autokosten 60

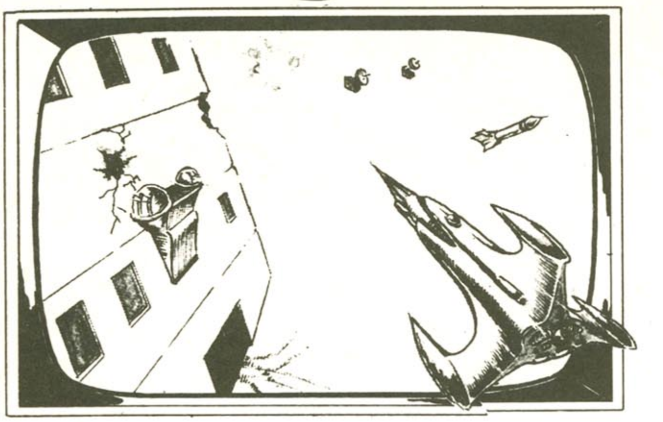
Texas 99

Dive-Bomber 57

Computerclub

Eine Regionalgruppe der A.U.G.E stellt sich vor 62

Tonight On Your Micro



CAN YOU SURVIVE PENETRATOR?

Penetrator is the most amazing and sophisticated arcade game yet devised for the 48K Spectrum with features never seen before! Cassette £6.95.



RESCUE A VIC PRINCESS

A multi-part adventure for the standard VIC 20! All you have to do is find the castle, defeat the dragon, find your way through the labyrinth, kill the wizard and rescue the Princess. And that's not half of it! The Wizard & The Princess cassette only £5.95.



CONVERT YOUR VIC INTO AN ARCADE MACHINE

Amazing arcade action for the standard VIC 20 with 5 new exciting games from Clifford Ramshaw: ALIEN BLITZ*, INVADERS*, GROUND ATTACK, STORM and SPACE ROCKS* (games marked * contain machine code). You must have the GAMES PACK cassette-only $5.95.



TALK TO THE SPECTRUM!

Why is this man smiling? You will too when you read Dr. Logan's book UNDERSTANDING YOUR SPECTRUM. The definitive guide for only £7.95.

MELBOURNE HOUSE PUBLISHERS

Orders to: 131 Trafalgar Road, Greenwich, London SE10

Correspondence to: Glebe Cottage, Station Road, Cheddington, Leighton Buzzard, BEDS LU7 7NA

HC 1

☐ Please send me your free catalogue.
Please send me:

Cassettes

16K SPECTRUM

- ☐ Over The Spectrum No. 1 — £5.95
- ☐ Over The Spectrum No. 2 — £5.95
- ☐ Over The Spectrum No. 3 — £5.95
- ☐ Programs from Spectrum Machine Language Book — £5.95

48K SPECTRUM

- ☐ Penetrator — £6.95
- ☐ The Hobbit — £14.95

STANDARD VIC 20

- ☐ VIC Innovative Cassette 1 — £5.95
- ☐ VIC Innovative Cassette 2 — £5.95
- ☐ VIC Innovative Cassette 3 — £5.95
- ☐ VIC Games Pack — £5.95
- ☐ The Wizard & The Princess — £5.95

Books

SPECTRUM

- ☐ Understanding Your Spectrum — £7.95
- ☐ Spectrum Machine Language For The Absolute Beginner — £6.95
- ☐ Over The Spectrum — £6.95

VIC 20

- ☐ VIC Innovative Computing — £6.95

All prices include VAT where applicable

£ ________

Please add 80p for post and pack £ .80

TOTAL £ ________

Unconditional Guarantee All Melbourne House cassette software is supplied on high quality cassettes with attractive inserts. Full instructions are provided and the cassettes are unconditionally guaranteed against malfunction.

I enclose my cheque/money order for £ ________

Please debit my Access card No. ________ Expiry date ________

Signature ________ Name ________

Address ________

________ Postcode ________

MELBOURNE HOUSE PUBLISHERS



Step into the future:
VISIT MIDDLE EARTH WITH YOUR SPECTRUM.

Melbourne House Publishers, in association with the Tolkien Estate, are very proud to announce the release of THE HOBBIT, a super-program like no other before, for the 48K Spectrum.

THE HOBBIT is totally unique, with features never before seen on any computer, let alone on the Spectrum.

Based on the fantasy land described in J. R. R. Tolkien's brilliant novel, in this program you take on the role of Bilbo, the hobbit: danger, adventure, and excitement are all part of it, presented to you in words and graphics, but it is you who must confront and solve the problems this time.

You instruct the computer in completely ordinary English sentences!

THE HOBBIT program is capable of very sophisticated communications, including features not previously thought possible on a microcomputer: adjectives, adverbs, multiple sentences, are just a part of the power of this program.

You will meet all your favourite characters from J. R. R. Tolkien's book, and amazing as it sounds, they will all have an independent life of their own. They can make decisions and act on them! They react not only to what you do, but also to every other character in THE HOBBIT!

Because of this unique feature, you will find that each time you play THE HOBBIT, events will proceed in a slightly different way, and the further you get into Tolkien's world, the more different each visit may be.

THE HOBBIT also allows you to talk to the other characters! At last you can answer the questions they may put to you, or more importantly, you can suggest to them what you would like them to do!

THE HOBBIT program follows closely the plot of the book, and so a copy of the book is included as part of THE HOBBIT package at no extra charge.

THE HOBBIT brings to you the future of microcomputers in an exciting and challenging fantasy. A super-program that is a milestone in computer software.

Other cassette software includes PENETRATOR, a 48K state-of-the- art arcade game – £6.95 including VAT.
Spectrum books include: Over The Spectrum – £6.95.
Spectrum Machine Language – £6.95.
Understanding Your Spectrum – £7.95.

Melbourne House Publishers HC 1
Orders to: 131 Trafalgar Road, Greenwich, London SE10.
Correspondence to: Glebe Cottage, Station Road, Cheddington, Leighton Buzzard, BEDS LU7 7NA.

☐ Please send me your **free** Spectrum Catalogue.

Please send me:
☐ The Hobbit package – 48K Spectrum cassette, full colour instruction manual and Hobbit book – £14.95
☐ Other ....

(Please add 80p for post & pack.) I enclose cheque/money order £.......... for my Access card No.......... Expiry date..........
Signature..........
Name ....
Address ....
....
.... Postcode..........

MELBOURNE HOUSE PUBLISHERS

Der Apple lernt sehen

Nicht immer müssen Firmen, die in der Computerbranche produzieren, größere Unternehmen sein. Daß sogar Einmann-Betriebe weltweite Aufmerksamkeit erzeugen können, zeigt unser Bericht.



Homecomputer (HC): Herr Bartscher, Ihre Video-Interfaces für den Apple II sind mittlerweile bis über unsere Grenzen hinaus bekannt. Wie kommt der Inhaber eines Rundfunk- und Fernsehfachgeschäftes dazu, für die Computerbranche zu produzieren?

Lothar Bartscher (L.B.): Das Ganze fing so an, daß mit Abschluß der Meisterprüfung 1977 der eigentlichen "Karriere" ein Ende gesetzt sein sollte, denn mehr, als die Meisterprüfung kann man im normalen Handwerk ja nicht machen. Also überlegte ich mir, was meine zukünftigen Lernobjekte sein sollten und durch irgendeinen Anstoß in meinem Freundeskreis, die beim Studium an die Microcomputerei gekommen waren und irgendeine Information aus den USA kam der Vorschlag, man könnte eigentlich einen Computer bauen und sich dadurch in die ganze Problematik des Computers einarbeiten.

Es wurde also der Entschluß gefaßt, keinen KIM zu kaufen, sondern, um das Projekt ordentlich zu studieren, den KIM nachzubauen.

Ich besorgte mir also die Manuals und ein paar Lochrasterplatten; die ICs wurden, weil sie damals in Deutschland noch nicht erhältlich waren, direkt in USA bestellt - und die kamen dann auch nach zirka 2 Monaten oder so, hier an.

Nach einem weiteren halben Jahr lief dann auch der KIM - nach vielen Rückschlägen. Aber trotz allem hieß es durchhalten und das hatte den Vorteil, daß man das System in und auswendig kennenlernte; was man bei fertig gekauften Systemen nie richtig erreichen kann.

HC: Aber an den Apple dachte man ja in Deutschland zu dieser Zeit noch gar nicht. Wie kamen Sie dann später auf die Idee, sich mit dem Apple zu beschäftigen?

L.B.: Ja, zunächst wurde erst einmal ein PET angeschafft, der damals gerade "im Kommen war"; das wird so etwa 1978 gewesen sein, oder Anfang 79?

HC: Ja, Mitte 1979 war der PET schon ziemlich gut auf dem deutschen Markt eingeführt.

L.B.: Die Entscheidung war wegen der vielen Ähnlichkeiten mit dem KIM leicht; selber Prozessor, gleiche Ports, usw.

Ich war nun einmal auf den 6502 eingestiegen und bin auch dabei geblieben.

Tja, und dann lief das alles so vor sich hin, mehr oder weniger war alles Spielerei.

Und dann, eines Tages kam ein alter Bekannter, den ich seit meiner Berufsschulzeit nicht mehr gesehen hatte in meinen Laden - ich hatte damals eine Apple-Werbung im Schaufenster und hatte geplant, neben Fernsehern auch Microcomputer anzubieten - dieser Bekannte sagte mir damals, er hätte sich einen Digitizer für den Apple zugelegt, wäre aber überhaupt nicht damit zufrieden, weil er ihm viel zu langsam war. Er brauchte so etwa 8 Sekunden, um ein Bild mittels einer Video-Kamera in den Apple zu laden.

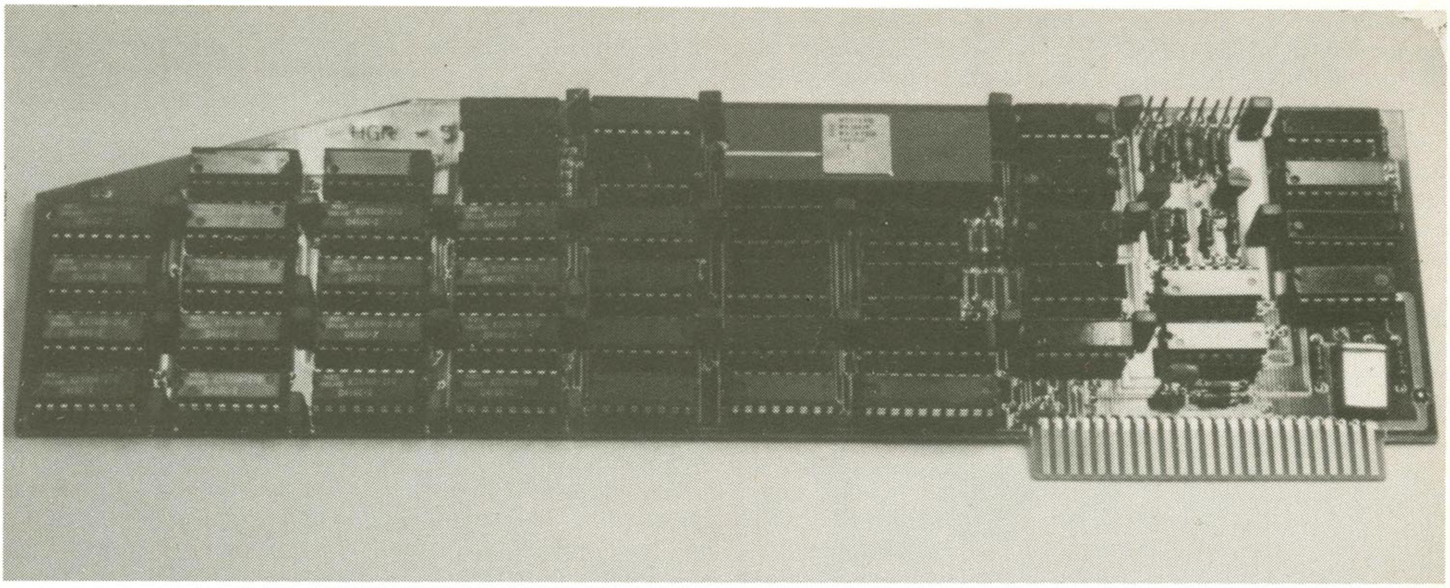
Er wolle Computerportraits machen und dazu bräuchte er natürlich ein Interface, das erheblich schneller ist, mindesten 1/25 Sekunde - und das gäbe es nicht.

HC: Und da haben Sie beschlossen, eines zu entwickeln, das schnell genug ist; das Bilder mit der Verschlußzeit einer normalen Kleinbildkamera in den Speicher des Apple lädt?

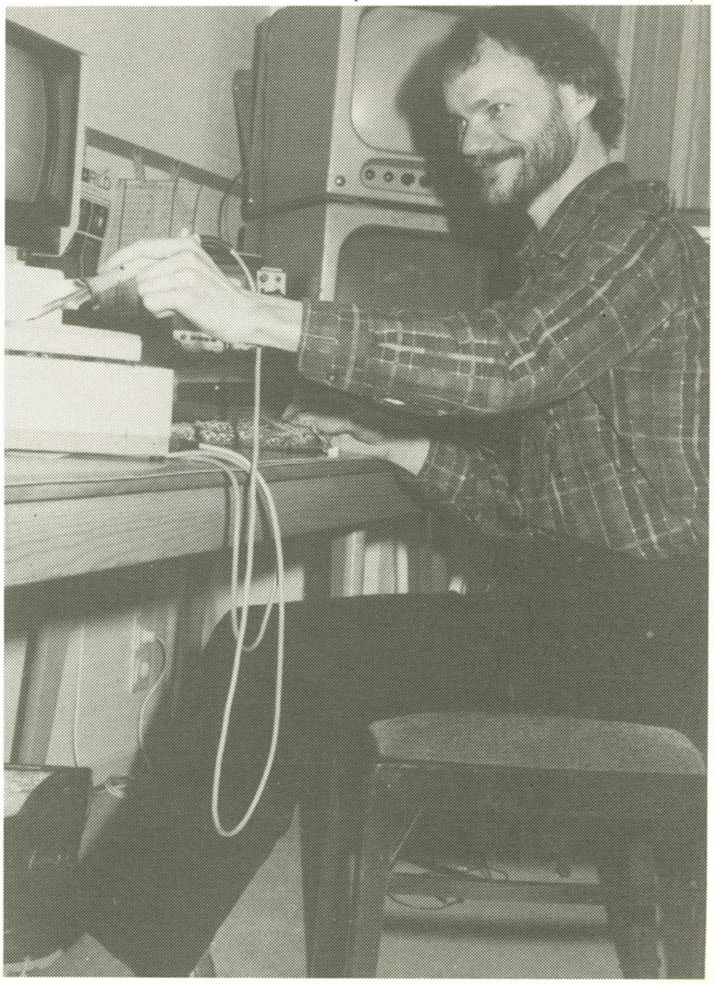
L.B.: Dann fing ich an, im stillen Kämmerlein Vorversuche zu machen und anfangs sah auch alles sehr einfach aus. Aber es verging, bis das erste Produkt lauffähig war ein ganzes Jahr.

HC: Sie sagen, am Anfang sah alles sehr einfach aus. Heißt das, daß Sie schon am Anfang Ihrer Entwicklung zu besseren Ergebnissen gekommen waren, als die zwei Hersteller amerikanischer Videokamera-Interfaces?

L.B.: Ja, eigentlich schon. Ich hatte schon nach meinen ersten Versuchen ein komplettes digitalisiertes Bild mit 16 Graustufen innerhalb von 1/50 Sekunde in den Apple gebracht, aber nicht jedesmal klappte es - und es stellte sich heraus, daß das System noch erhebliche Fehler hatte. Zudem mußte



jedesmal das gesamte System neu konstruiert werden, weil sich zeigte, daß die Fehler nicht anders zu beseitigen waren.
Aber schließlich, nach einem Jahr waren alle Probleme gelöst und der Prototyp dieser Platine stand, konnte zum Layouter gehen und die ersten Vorführungen konnten über die Bühne gehen.
Man kann sagen, daß, wenn es nicht das Schnellste sein sollte, so gehört es zumindestens zu den schnellsten Videokamera-Interfaces.
HC: Uns ist keines bekannt, das 16 Graustufen innerhalb von einer fünfzigstel Sekunde in der Auflösung im Applespeicher unterbringt und wir haben in Vorbereitung des Gesprächs mit Ihnen alle bekannten Interfaces getestet.
L.B.: Als nächstes entwikkelte ich ein Interface, das weit mehr Punkte in den Speicher schob, als der Apple auf seinem Display darstellen kann. Das heißt, ich bringe 32 K-Byte Bildinformation zur Auswertung in den Speicher; der Apple kann aber nur 8 K darstellen.
Plötzlich war also der Punkt erreicht, an dem man die Leistungsfähigkeit des Apple, als Display, überrundet hatte. Der Punkt wurde mit etwas Frustration abgeschlossen und mehr oder weniger in die Schublade gelegt, denn warum soll man ein Interface bauen, was soviel leistet, daß es der Computer dann gar nicht ausnutzen kann.
Dann kam ich auf die Idee, man müßte eigentlich als Ergänzung dazu, um die Leistungfähigkeit des Interfaces darzustellen, eine noch höher-auflösende Grafik für den Apple entwickeln, als er ohnehin schon hat.
Der Apple ist ja bekannt geworden, aufgrund seiner Grafikstärke - aber warum soll man nicht was gut ist, noch weiter verbessern.
Das Resultat war dann die Steckkarte HGA-512. Karten dieser Art, die 512 mal 512 Punkte auflösen konnten waren damals schon im Gespräch, allerdings nicht als Steckkarten für den Apple, sondern man mußte erst anfangen entsprechende Adapter zu bauen, diese an den Apple-Bus anpassen und das will ja der End-User in den meisten Fällen nicht machen; er. will eine Karte 'reinstecken' und anfangen zu programmieren. Es wurde deshalb auch hier wieder eine fertige Karte hergestellt, die nicht nur die 512 mal 512 Punkte in Schwarz/Weiß darstellen kann, sondern durch einen Poke-Befehl umschaltbar auf 256 mal 256 Punkte und 16 Graustufen auf jedem Punkt ist.
HC: An jedem Punkt 16 Graustufen oder 16 Farben, wie man auf diesem Monitor sehen kann?
L.B.: An einem RGB-Monitor mit 16 Farben, die dann ganz willkürlich jeder Graustufe zugeordnet sind. Und zusammen mit dieser Karte war jetzt eine Digitalisierung möglich, die, wenn man ein Foto davon macht, von einem direkt aufgenommenen Foto nicht mehr zu unterscheiden ist.
Und das ist auch der momentane Stand. Und auch die Spezialisierung in die wir uns hineinbegeben werden, sämtliche Arten von Bildver-



Im Gespräch

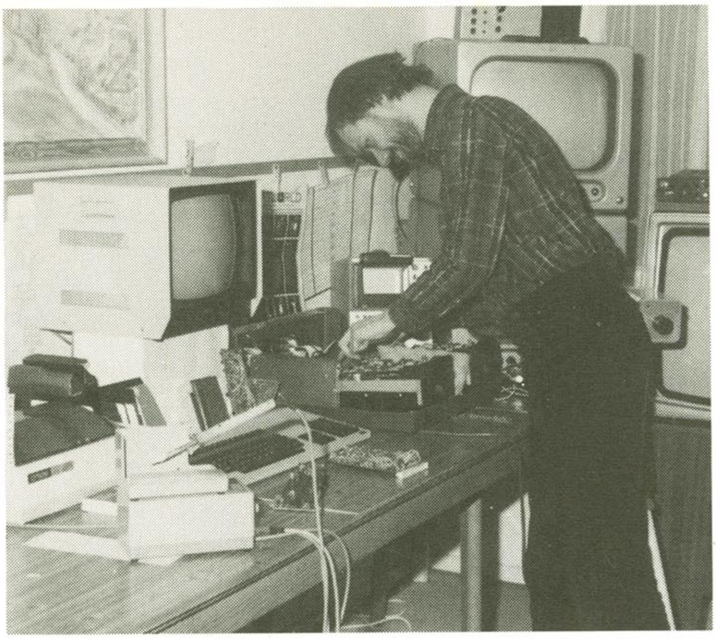
arbeitung, die man durch ein preisgünstiges Steuergerät, das ja der Apple darstellt, möglich machen kann.

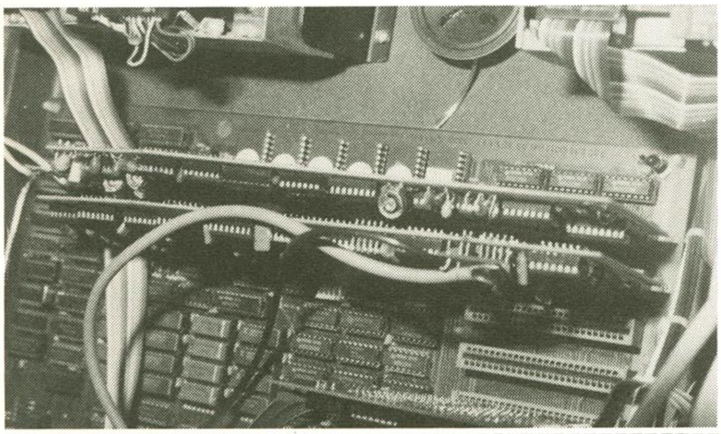
HC: Und wo liegen die Anwendungsbereiche dieser Video-Karten, außer den bereits angesprochenen Computerportraits?

L.B.: Anwendungsbereich; das beginnt bei einer einfachen Analyse, die man mit einer Kamera macht, wo man ein Bild nach seinen Grauwerten abfragt über Anwendungen; zum Beispiel in einer Arztpraxis, in der ein Arzt - in Verbindung mit dem Graphicstablet - Auswertungen von Ultraschallaufnahmen vornimmt, indem er auf den digitalisierten Bildern Flächen umfährt und sich die Inhalte ausrechnen läßt.

Überhaupt, für Bildanalysen aller Art ist dies der preisgünstigste Einstieg, den man sich vorstellen kann.

HC: Beim lieben Geld angekommen, was kostet so ein Sytem zur Zeit?





L.B.: Also zur Zeit kosten beide Interface-Karten je 1356,-DM, wobei das HGR-Interface einen extra Grafikprozessor enthält und 32 K Speicher, sodaß noch nicht einmal die HGR-Page im Apple benötigt wird.

HC: Herr Bartscher, wir wünschen Ihnen weiterhin viel Erfolg und viele Ideen für die Zukunft.

Schönen Dank für das Gespräch.

VC-64 lernt sprechen

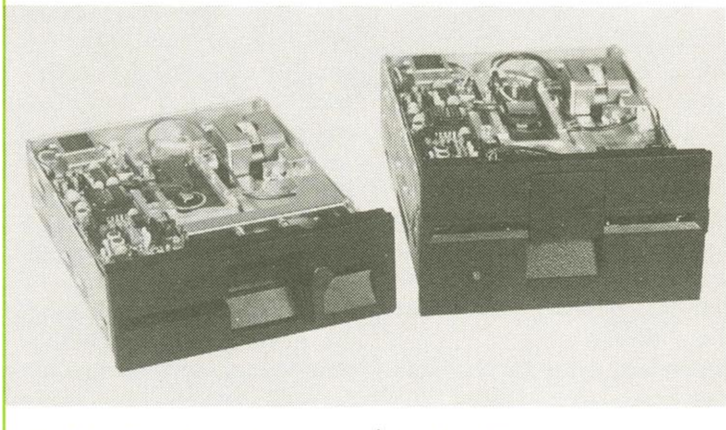
Als vor einigen Jahren der Microcomputerboom begann, konnte niemand auch nur ahnen, was innerhalb kürzester Zeit machbar sein würde.
Am Anfang waren einfache Ballspiele, später kamen bewegte Grafiken und aufwendige Geräuschkulissen hinzu.
Noch in diesem Jahr kommt neue fortschrittliche Technologie in unsere Homecomputer: Der Heimprogrammierer kann in seine Programme in Kürze Sprache mittels Basic-Kommandos einbauen.
In der Preisklasse um 200 DM soll der neue Sprach-Synthesizer für den VC-64 bis Mai/Juni lieferbar sein.
Entwickelt wurde diese Zusatzcartridge, die einfach in den Rom-Slot gesteckt wird, von Commodore Speech Technology in Dallas. Sie ist in der Lage, eine Anzahl von verschiedenen Stimmen, männlich, weiblich, bis hin zur Donald-Duck-Stimme zu erzeugen.
Eine Wortfolge sprechen zu lassen, geschieht von der Tastatur oder vom Programm aus mit dem Basic-Befehl SAY und dem auszugebenden Satz in Anführungszeichen - also anstelle von PRINT schreibt man SAY, und der Satz erscheint nicht in Schriftform auf dem Bildschirm, sondern tönt aus dem Lautsprecher des Fernsehers.
Unseren Informationen zufolge, wird dieser neue Sprach-Synthesizer auch auf der Hannover-Messe zu hören sein.

Super-Printmaster 3

Ein neues fortschrittliches Drucker-Interface für den Apple II.

Textverarbeitung und Graphik, beides in 16 Farben an einen Colordrucker zu übertragen war bisher nur mit Apple-RAM-fressender Software zu erreichen. Mit dem Super-Printmaster II wird es kinderleicht und verbraucht nicht 1 Byte RAM. Einfachste Kommandos genügen, um Texte, Balkengrafiken und digitalisierte Bilder in 16 verschiedenen Farben in bis zu 16-facher Vergrößerung zu übertragen. Neben einer Menge eingebauter Grafik-Text-Möglichkeiten wird das farbige Ausdrucken der Hi- und Lowresolution-Bildschirme des Apple besonders erleichtert.
Kontaktadresse:
Digitek (International) Ltd, 37c West Street, Horsham, West Sussex, RH12 1PP, England

"SLIMLINE"-Varianten



Nicht jeder Anwender ist in der Lage, von heute auf morgen auf die modernen 5 1/4" Floppy-Disk-Laufwerke mit 41 mm Höhe zu wechseln.
TEAC hat für seine "SLIMLINE"-Laufwerke, Serie FD-55, Frontplatten entwickelt, die sowohl der 2/3 Bauhöhe (BASF, Philips, Shugart, Tandon usw.) entsprechen.
Der Anwender ist somit in der Lage, ohne Änderung des vorhandenen Designs, Laufwerke in modernster Technologie einzusetzen. So wird bei TEAC z.B. ein bürstenloser Gleichstrommotor als Direktantrieb eingesetzt. Dies gewährleistet sowohl einen exakten Gleichlauf, als auch einen verschleißfreien Betrieb.
Die TEAC-Laufwerke sind in 40- und 80-Spur Ausführung, seitig und doppelseitig, von 250 KByte bis 1 MByte lieferbar.
Info:
abo ELEKTRONIK, Herrsching

TS2000 - der USA-Spectrum



Der Sinclair Spectrum und der ZX-Drucker haben von der Firma Timex für den amerikanischen Markt ein neues Gewand und neue Namen erhalten.
TS2000 heißt der Spectrum in Amerika und TS2040 der Drucker.
Gleichzeitig wurde der ZX-81 - die Fa. Timex verkauft ihn unter dem Namen TS1000 im Preis um ca. 40 DM gesenkt.
Auch die TS2000- & TS2040-Preise können sich sehen lassen. Sie betragen etwa 360,-DM für die 16K-Version und um 500,-DM für die 48K-Ausstattung. Der Drucker wird in den Staaten für etwa 220,-DM angeboten.

News & Trends

VC-20 weltweit Nr.1

Commodore vorn



Zum Jahresende von der Firma Commodore veröffentlichte Zahlen weisen für das Geschäftsjahr bis Juni 82 Umsätze von über 800 Mio DM aus - eine Steigerung von 63 % gegenüber dem Vorjahr. Dabei wurden ca 120 Mio DM Gewinne erzielt.
75 % davon ist auf die Verkaufszahlen von den bekannten Microcomputern des Unternehmens zurückzuführen. Während sich die CBMs der PET-Klasse weiterhin gut verkauften, hat sich vor allem der VC-20 als absoluter Renner ausgezeichnet. Über 800 000 Stück wurden allein in den USA davon abgesetzt und machten ihn zum weltweit meistverkauften Microcomputer.
Während in Amerika der Verkauf von Computern Zuwachsraten von 190 % im letzten Jahr erreichte, stieg der Verkauf in der übrigen Welt immerhin um ca 34 %.

E.T.A.

... der Außerirdische

Science-Fiction-Filme üben einen besonderen Reiz auf Spiele-Autoren, besonders auf Computerspiele-Hersteller aus.
Nicht umsonst wimmelt es nur so von Spocks, Darth Vaders, Trons und ETs.
Nach Atari - immer mit der besten Spürnase auf diesem Gebiet - erreichen nun die ETs anderer Software-Firmen die gute alte Erde.
Eine interessante Version erreichte uns für den Sinclair Spectrum von der Firma Automata, Southsea, England.
Nach einer Bruchlandung hat ET seinen Namen vergessen, sucht sein Raumschiff, und ist auf der Flucht vor Hunden und Soldaten.
Der Spieler muß den Namen finden, und die Agriffe auf ET verhindern.
Das Spiel kostet, wie die meisten Spiele für den Spectrum, umgerechnet weniger als 20,-DM.

Galaxians & Defenders

Die neue Welle rollt

Fast waren sie schon vergessen; doch die alten Spielhallen-Klassiker Galaxians und Defenders sind wieder da.
Atari, für die Maschinen Atari 400 und 800, sowie Acornsoft für den BBC Model B haben Neuauflagen soeben auf den Markt gebracht. Während sie bei Acornsoft unter neuen Titeln (Arcadians, Planetoids) rangieren behielten sie bei Atari die bekannten Namen bei.

Gelegenheit ergreifen

Dornröschen-Schlaf endlich beenden!

Seit 1975, als Steve Woniak in einer kleinen californischen Werkstatt die Firma Apple gründete, hat sich der Microcomputermarkt in den USA auf geschätzte 10 Milliarden DM Jahresumsatz gemausert.
Innerhalb der letzten eineinhalb Jahre ist dort die Zahl der Microcomputerhersteller von circa 50 auf über 150 angewachsen. Englische Firmen, wie Sinclair, Tangerine, Bug-Byte und Quicksilva, um nur einige zu nennen exportieren weltweit in riesigen Mengen (z.B.: mehr als 750 000 ZX-81 arbeiten in aller Welt).
Überall stellt man sich auf die neuen Technologien ein, erkennt das riesige Potential an begeisterten Heimcomputer-Anwendern.
Und in Deutschland?
Hier schlafen sich, von wenige Ausnahmen einmal abgesehen, die dafür in Frage kommenden Firmen regelrecht zu Tode. Außer ein paar Apple-Nachbauten ist kaum nennenswertes zu vermelden.
Seit Jahren schwappt eine Welle nach der anderen, besonders im Freizeitbereich aus den Staaten über unser Land. Immer mit der regelmäßigen Verspätung von 5 Jahren (man denke nur an Rock 'n' Roll, Hoola-Hoop, Skateboards, Rollerskates und neuerdings Aerobic).
Immer und immer wieder erleben wir es, daß 'schlaue' Köpfe in den Medien uns diese Sachen jahrelang ausreden wollen, bis es uns dann um so schlimmer erwischt - mit dem Unterschied, daß dann die Anderen das Geschäft machen.
Der Heimcomputermarkt ist, das kann eigentlich jeder, der sich damit beschäftigt, erkennen, eine neue große Herausforderung für unsere Industrie, wahrscheinlich die größte, die es je gab, denn dieser Bereich schneidet weit mehr in unser aller Leben ein, als alle oben genannten Spielereien.
Nimmt man hier die Herausforderung an? Bis jetzt noch nicht. Aber es wird Zeit, endlich aufzuwachen, bevor es zu spät ist. Denn der Microcomputer ist keine vorübergehende Modeerscheinung wie die obengenannten Freizeitwellen, sondern, ähnlich wie das Fernsehen es heute ist, ein auf Dauer lebensbestimmender Faktor, an dem niemand vorbeikommenen wird.

Biorhythmus-Programm für VC-64

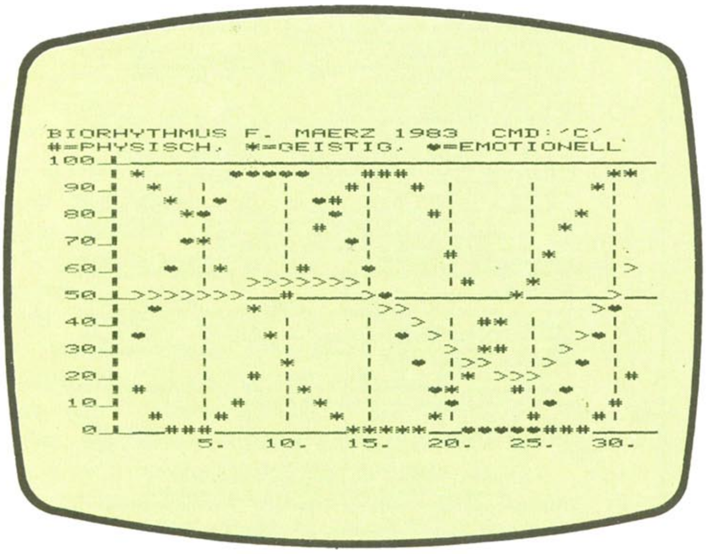
Spätestens mit Erscheinen programmierbarer Taschenrechner begann der Siegeszug der Biorhythmik. Durch den Microcomputer, nebst Drucker oder Plotter wurde daraus sogar ein einträgliches Geschäft.

Tatsächlich sind Programme im Handel, für die - man kann es kaum glauben 250,-DM und mehr verlangt werden. 250,-DM für eine einfache Sinuskurvenberechnung - um mehr handelt es sich ja schließlich nicht - und ein Kalenderprogramm.

Für den VC-64 ist bisher noch kein Biorhythmus-Programm erschienen; drucken wir also in Homecomputer unsere Version zum altbekannten Thema ab.

Farbige Darstellung ist mit dem VC-64 besonders schön zu programmieren, weil man aus 16 Farben wählen kann, die sich mit 16 Hintergrundfarben kombinieren lassen.

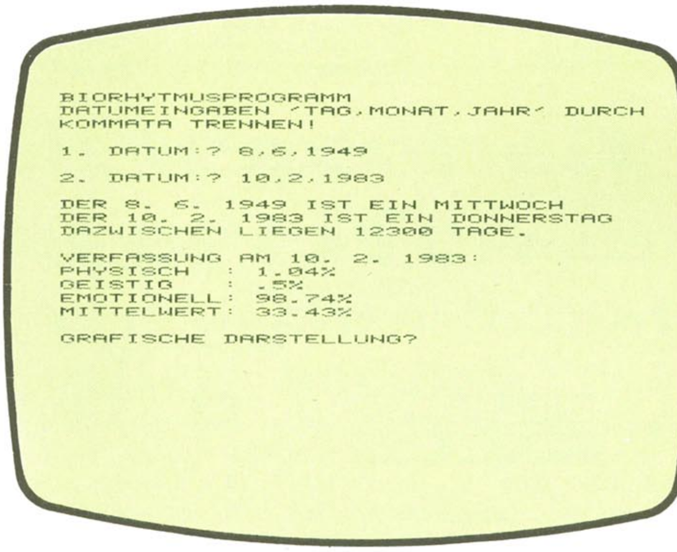
Benutzt man außerdem einige der bekannten Commodore-Graphik-Zeichen, dann hat man auch die Möglichkeit, alles auf einen Drucker auszugeben.



So sieht ein Ausdruck des Bildschirms aus. Die Linie aus den Größer-als-Zeichen zeigt den Mittelwert an, der sich aus den Kurven errechnet.

Zur Anwendung des Programms ist noch festzustellen, daß mit <P> ein Ausdruck des Bildschirmes auf einen Drucker, der fähig ist die PET-Zeichen zu drucken, ermöglicht wird. Nach Drücken der Taste <C> werden die Kurven für den nächsten Monat dargestellt.

Die Taste <SPACE> dient zum Neustarten des Programms, d.h. sie wird gedrückt, wenn mit neuen Daten gearbeitet werden soll.



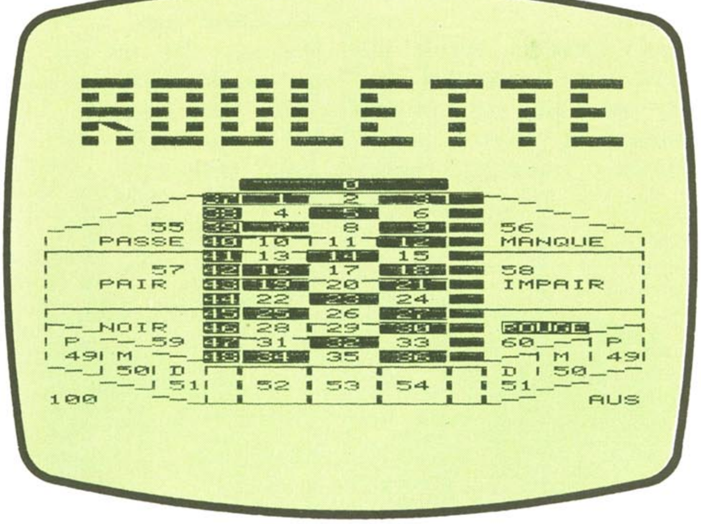

```
10 POKE53280,7:POKE53281,7: PRINT"BIO
RHYTMUSPROGRAMM"
20 PRINT"DATUMEINGABEN 'TAG,MONAT,JAHR'
 DURCH      KOMMATA TRENNEN!"
30 VB=1024:VC=55296
40 DIMB$(6),C$(12)
50 DATA"MONTAG","DIENSTAG","MITTWOCH","D
ONNERSTAG","FREITAG","SAMSTAG","SONNTAG"
60 DATA"JANUAR","FEBRUAR","MAERZ","APRIL
","MAI","JUNI"
70 DATA"JULI","AUGUST","SEPTEMBER","OKTO
BER","NOVEMBER","DEZEMBER"
80 FORI=0TO6:READB$(I):NEXT:FORI=1TO12:R
EADC$(I):NEXT
110 INPUT"1. DATUM:";GT,GM,GJ
120 INPUT"2. DATUM:";HT,HM,HJ
140 T=GT:M=GM:J=GJ:GOSUB1000
150 GS=S:GD=J:GW=WT
160 T=HT:M=HM:J=HJ:GOSUB1000
170 HS=S:HD=J:HW=WT:JS=HS-GS
180 PH=(SIN(JS*360/23*π/180)+1)*50:GE=(S
IN(JS*360/33*π/180)+1)*50
185 SE=(SIN(JS*360/28*π/180)+1)*50
190 MW=(PH+GE+SE)/3
197 IFGW=6THENB$="SONNTAG"
200 PRINT"DER";GT;".";GM;".";GJ;"IST
EIN ";B$(GW)
220 PRINT"DER";HT;".";HM;".";HJ;"IST E
IN ";B$(HW)
225 PRINT"DAZWISCHEN LIEGEN";JS;"TAGE."
230 IFGJ>1582ANDGJ<1700THENSU=47.25:GOTO
260
231 IFGJ>1699ANDGJ<1800THENSU=46.25:GOTO
260
232 IFGJ>1799ANDGJ<1900THENSU=45.25:GOTO
260
233 IFGJ>1899ANDGJ<2100THENSU=44.25:GOTO
260
234 IFGJ>2099ANDGJ<2200THENSU=43.25:GOTO
260
235 IFGJ>2199ANDGJ<2300THENSU=42.25:GOTO
260
236 IFGJ>2299ANDGJ<2500THENSU=41.25:GOTO
260
240 IFGJ=1582ANDGM>9THEN250
242 PRINT"AUSSERHALB DES BEREICHS '15.1
0.1582' BIS"
243 PRINT"31.12.2499 IST KEINE BERECHNU
NG DES BIO-RHYTHMUS MOEGLICH!"
244 GOSUB2000
245 GOTO245
250 IFGM=>9ANDGT>14THENSU=47.25
251 IFGM>10THENSU=47.25
260 PRINT"VERFASSUNG AM";HT;".";HM;".
";HJ;":"
261 PRINT"PHYSISCH   :";INT(PH*100)/100;"
%"
262 PRINT"GEISTIG    :";INT(GE*100)/100;"
%"
263 PRINT"EMOTIONELL:";INT(SE*100)/100;"
%"
264 PRINT"MITTELWERT:";INT(MW*100)/100;"
%"
270 PRINT"GRAFISCHE DARSTELLUNG?"
280 GETA$:IFA$="J"THEN300
290 IFA$="N"THEN120
292 IFA$="P"THENGOSUB3000
295 GOTO280
```

```
300 PRINT"□BIORHYTHMUS F. ";C$(HM);HJ;"
CMD:"C/"
320 PRINT"▮*=PHYSISCH, ▮▮*=GEISTIG, ▮▮*=EM
OTIONELL"
340 PRINT"□100_▌________________________________
_____________"
341 PRINT"      ▌ "
342 PRINT" 90_▌    |    |    |    |    |
|"
343 PRINT"      ▌    |    |    |    |    |
|"
344 PRINT" 80_▌    |    |    |    |    |
|"
345 PRINT"      ▌    |    |    |    |    |
|"
346 PRINT" 70_▌    |    |    |    |    |
|"
347 PRINT"      ▌    |    |    |    |    |
|"
348 PRINT" 60_▌    |    |    |    |    |
|"
349 PRINT"      ▌    |    |    |    |    |
|"
350 PRINT" 50_▌________________________________
_____________"
351 PRINT"      ▌ "
352 PRINT" 40_▌    |    |    |    |    |
|"
353 PRINT"      ▌    |    |    |    |    |
|"
354 PRINT" 30_▌    |    |    |    |    |
|"
355 PRINT"      ▌    |    |    |    |    |
|"
356 PRINT" 20_▌    |    |    |    |    |
|"
357 PRINT"      ▌    |    |    |    |    |
|"
358 PRINT" 10_▌    |    |    |    |    |
|"
359 PRINT"      ▌    |    |    |    |    |
|"
360 PRINT"  0_▌________________________________
_____________"
361 PRINT"          5.   10.   15.   20.   25
.  30."
370 HT=0
400 HT=HT+1
410 T=HT:M=HM:J=HJ:GOSUB1000
420 HS=S:HD=J:HW=WT:JS=HS-GS
430 PH=(SIN(JS*360/23*π/180)+1)*50:GE=(S
IN(JS*360/33*π/180)+1)*50
440 SE=(SIN(JS*360/28*π/180)+1)*50
450 MW=(PH+GE+SE)/3
460 IFHT>27THEN600
470 SP=INT(HT+4):E=PH:GOSUB2000
480 POKE(VC+40*ZE+SP),5:POKE(VB+40*ZE+SP
),35:E=GE:GOSUB2000
490 POKE(VC+40*ZE+SP),6:POKE(VB+40*ZE+SP
),42:E=SE:GOSUB2000
500 POKE(VC+40*ZE+SP),10:POKE(VB+40*ZE+S
P),83:E=MW:GOSUB2000
505 POKE(VC+40*ZE+SP),0:POKE(VB+40*ZE+SP
),62
510 GOTO400
600 T=1:M=HM+1:IFM=13THENM=1:J=HJ+1:GOTO
620
610 J=HJ
620 GOSUB1000
630 IFHW=WTTHEN700
640 GOTO470
700 HM=HM+1:IFHM=13THENHM=1:HJ=HJ+1
710 GETA$:IFA$="  "THENRUN
712 IFA$="P"THENGOSUB3000
715 IFA$="C"THEN300
720 GOTO710
1000 IFM<3THENJ=J-1:M=M+12
1010 S=INT(365.25*J)-INT(J/100)+INT(J/40
0)+31*(M-1)-INT(.4*M+2.3)+T
1020 J=S+1721060:WT=J-INT(J/7)*7:RETURN
1100 K=INT((JS-PV)/PZ+1)
1110 TE=GS+PZ*K+PV:JX=INT((TE-SU)/365.25
)
1120 TN=INT(JX*365.25)-INT(JX/100)+INT(J
X/400)+60-TE
1130 IFTN<0THENTN=ABS(TN)
1140 IFTN>0THENTN=TN*-1
1150 MX=INT((TN+92.3)/30.6)
1160 TX=TN-31*(MX-1)+INT(.4*MX+2.3)+60
1170 IFMX>12THENMX=MX-12:JX=JX+1
1180 RETURN
2000 ZE=22-INT(E/5)
2010 IFZE=2THENZE=3
2020 RETURN
3000 OPEN4,4:PRINT#4,CHR$(14)
3005 FORI=VBTOVB+999STEP40:P$=""
3010 FORJ=0TO39:X=PEEK(I+J)
3020 IFX<32THENX=X+64:GOTO3100
3025 IFX<64THEN3100
3030 IFX<96THENX=X+32:GOTO3100
3035 IFX<128THENX=X+64:GOTO3100
3100 P$=P$+CHR$(X)
3150 NEXT:PRINT#4,P$:NEXT
3155 CLOSE4
3160 RETURN

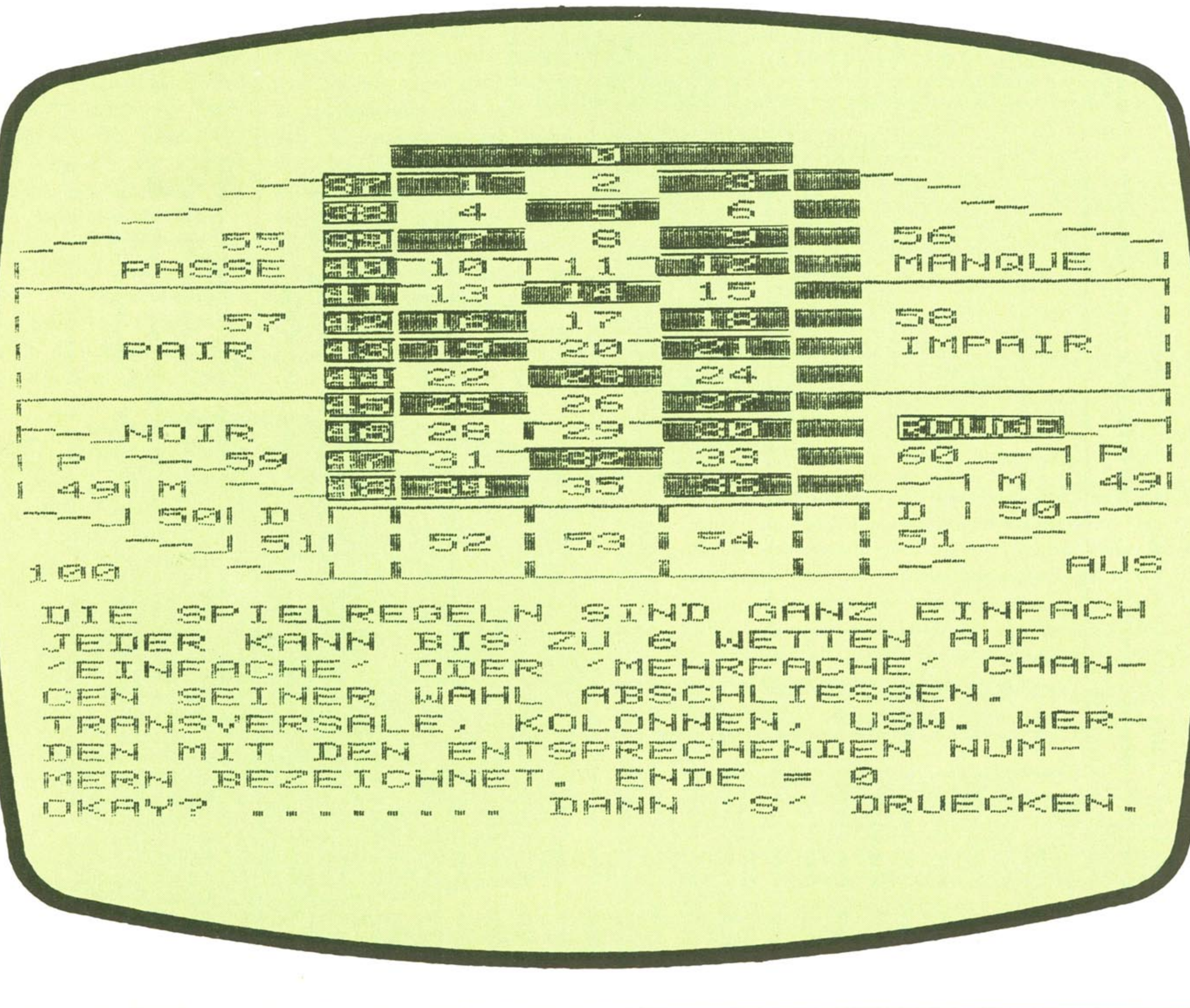
READY.
```

Roulette
für VC-64 & PET

Echte Spielbankathmosphäre vermittelt diese Simulation eines Roulettespiels, bei dem sämtliche Details originalgetreu implementiert wurden.



Jeder Spieler hat die Möglichkeit 6 mal während einer Runde seine Einsätze zu tätigen, was bedeutet, daß er alle im Originalspiel möglichen Einsatzvariationen nachvollziehen kann.
Daß das Spiel, wenn der VC-64 an einen Farbfernseher angeschlossen ist, besonders ansprechend aussieht, soll nur am Rande erwähnt sein.
Ergänzend zu der abgedruckten Spielregel (Bild unten) sei noch das Aussteigen erklärt. Der Spieler, der das Spiel beenden möchte setzt einfach sein gesamtes Guthaben auf die Zahl 100. Damit wird er vom Computer verabschiedet.
Das Glücksrad dreht sich in der Originalreihenfolge, das heißt die Farben und Zahlen stimmen mit denen der Original-Roulette-Tische überein. Die Zeit, die vergeht bis die simulierte Kugel zum Stillstand kommt wird allerdings von einer errechneten Zufallszahl bestimmt.



```
5 POKE53280,13:POKE53281,13
10 PRINT"
11 PRINT"
12 PRINT"
13 PRINT"
14 PRINT"
15 PRINT" "
16 PRINT" "
101 PRINT"                    0         "
102 PRINT"            37  1   2   3
103 PRINT"            38  4   5   6
104 PRINT"       55   39  7   8   9
 56 "
105 PRINT"  I   PASSE   40  10  11  12
 MANQUE  I "
106 PRINT"            41  13  14  15
107 PRINT"  I     57  42  16  17  18
 58   I "
108 PRINT"  I   PAIR   43  19  20  21
 IMPAIR  I "
109 PRINT"  I          44  22  23  24
110 PRINT"             45  25  26  27
111 PRINT"    NOIR   46  28   29
30    ROUGE   "
112 PRINT"  I P   59   47  31  32  33
 60   P I "
113 PRINT"  I 49I M   48  34  35  36
  M I 49I "
114 PRINT"    J 50I D I I I I I D
 I 50   "
115 PRINT"    J 51I I 52 I 53 I 54 I I 5
1   "
116 PRINT"   100
 AUS"
200 I=0:DIMRA$(37):DIMCO(37,7):DIMNA$(8)
210 FORJ=1TO37:FORK=1TO7:READCO(J,K):NEX
TK,J
220 DATA0,0,0,0,0,57,59,32,47,51,53,55,5
7,60,15,41,50,54,56,58,59,19,43,50,52,55
230 DATA58,60,4,38,49,52,56,57,59,21,43,
50,54,55,58,60,2,37,49,53,56,57,59,25,45
240 DATA51,52,55,58,60,17,42,50,53,56,58
,59,34,48,51,52,55,57,60,6,38,49,54,56
250 DATA57,59,27,45,51,54,55,58,60,13,41
,50,52,56,58,59,36,48,51,54,55,57,60,11
260 DATA40,49,53,56,58,59,30,46,51,54,55
,57,60,8,39,49,53,56,57,59,23,44,50,53
270 DATA55,58,60,10,40,49,52,56,57,59,5,
38,49,53,56,58,60,24,44,50,54,55,57,59
280 DATA16,42,50,52,56,57,60,33,47,51,54
,55,58,59,1,37,49,52,56,58,60,20,43,50
290 DATA53,55,57,59,14,41,50,53,56,57,60
,31,47,51,52,55,58,59,9,39,49,54,56,58
300 DATA60,22,44,50,52,55,57,59,18,42,50
,54,56,57,60,29,46,51,53,55,58,59,7,39
310 DATA49,52,56,58,60,28,46,51,52,55,57
,59,12,40,49,54,56,57,60,35,48,51,53,55
320 DATA58,59,3,37,49,54,56,58,60,26,45,
51,53,55,57,59
600 MO(1)=36:MO(2)=12:MO(3)=3:MO(4)=3:MO
(5)=2:MO(6)=2:MO(7)=2
610 DATA"  0 ","  32 ","  15 ","  1
9 "
615 DATA"  4 ","  21 ","  2 ","  2
5 ","  17 "
620 DATA"  34 ","  6 ","  27 ","  1
3 "
625 DATA"  36 ","  11 ","  30 ","  
8 ","  23 "
630 DATA"  10 ","  5 ","  24 ","  
16 "
635 DATA"  33 ","  1 ","  20 ","  1
4 ","  31 "
640 DATA"  9 ","  22 ","  18 ","  2
9 "
645 DATA"  7 ","  28 ","  12 ","  3
5 ","  3 "
650 DATA"  26 "
660 FORJ=1TO37:READRA$(J):NEXT
750 PRINT"DIE SPIELREGELN SIND GAN
Z EINFACH"
760 PRINT"JEDER KANN BIS ZU 6 WETTE
N AUF"
770 PRINT"'EINFACHE' ODER 'MEHRFACH
E' CHAN-"
780 PRINT"CEN SEINER WAHL ABSCHLIES
SEN."
790 PRINT"TRANSVERSALE, KOLONNEN, U
SW. WER-"
800 PRINT"DEN MIT DEN ENTSPRECHENDE
N NUM-"
801 PRINT"MERN BEZEICHNET. ENDE = 0
"
802 PRINT"OKAY? ........ DANN 'S' D
RUECKEN."
803 GETA$:IFA$><"S"THEN803
810 I=I+1
815 GOSUB5000
817 F1=1
818 IFI>8THENI=8:GOTO2300
820 PRINT"";I;". M
ITSPIELER, NAME";TAB(10):INPUTNA$(I)
830 KT(I)=1000
840 IFI=8THEN2300
850 PRINT"NOCH JE
MAND?                 "
860 GETA$:IFA$="J"THEN810
870 IFA$><"N"THEN850
880 GOTO2300
1000 GOSUB5000
1005 PRINT""
1010 FORX=1TOI
1020 PRINTTAB(4);X;". ";NA$(X);TAB(20);"
 HAT";KT(X);"$"
1030 NEXT
1040 T=TI
1050 IF(TI-T)<300THEN1050
1052 GOSUB5000
1053 IFF1=1THEN1058
1054 PRINT"MOECH
TE NOCH JEMAND EINSTEIGEN?"
1055 GETA$:IFA$="J"THEN810
```

```
1056 IFA$><"N"THEN1055
1058 F=0:F1=0
1060 FORJ=1TO6
1070 FORX=1TOI
1075 IFGE(X,J-1)=0ANDF>0THEN1100
1080 GOSUB5000
1090 GOSUB5100
1100 GE(X,J)=0:FE(X,J)=0
1150 IFF=0THEN1300
1200 IFGE(X,J-1)=0THEN1500
1300 IFKT(X)=0THEN1500
1400 PRINT"";NA
$(X);",";"  ( KTO=";KT(X);"$)"
1405 INPUT"WIEVIEL DOLLAR ";GE(X,J)
1408 IFGE(X,J)=0THENFE(X,J)=0:GOTO1500
1410 INPUT"AUF WELCHES FELD";FE(X,J
)
1412 IFFE(X,J)=100THEN4000
1415 IFFE(X,J)=0ORFE(X,J)>60THEN1430
1418 KT(X)=KT(X)-GE(X,J)
1420 IFGE(X,J)>1ORGE(X,J)=0THEN1440
1430 PRINT"NICHT GESTATTET!":T=TI
1431 KT(X)=KT(X)+GE(X,J)
1432 IF(TI-T)<180THEN1432
1434 GOSUB5000
1436 GOSUB5100
1438 GOTO1300
1440 IFKT(X)=>0THEN1500
1450 PRINT"KONTO IST UEBERZOGEN!"
1460 KT(X)=KT(X)+GE(X,J):GE(X,J)=0:FE(X,
J)=0
1470 PRINT"KONTOSTAND=";KT(X):T=TI
1480 IF(TI-T)<240THEN1480
1490 GOSUB5000
1495 GOSUB5100
1497 GOTO1300
1500 NEXTX
1510 F=1
1520 NEXTJ
1530 GOSUB5000
1540 PRINT"R
IEN NE VA PLUS!"
1550 T=TI:GOTO1600
1560 IFRND(TI)>.9955THEN2000
1600 RA=RA+1:IFRA=38THENRA=1
1610 PRINT"";RA$(RA
)
1620 IF(TI-T)>300THEN1560
1630 GOTO1600
2000 GOSUB5000
2020 IFCO(RA,7)=60THENWO$="ROUGE,  ":GOTO
2040
2030 WO$="NOIR,  "
2040 IFCO(RA,6)=58THENWO$=WO$+"IMPAIR,  "
:GOTO2060
2050 WO$=WO$+"PAIR,  "
2060 IFCO(RA,5)=56THENWO$=WO$+"MANQUE":G
OTO2100
2070 IFCO(RA,5)=55THENWO$=WO$+"PASSE"
2100 PRINT""
;WO$
2102 T=TI
2104 IF(TI-T)<300THEN2104
2110 FORX=1TOI
2120 FORN=1TO7
2130 FORJ=1TO6
2140 IFFE(X,J)=CO(RA,N)THEN2160
2150 GOTO2200
2160 IFRA=1THENGE(X,J)=GE(X,J)/2:GOTO218
0
2170 GE(X,J)=GE(X,J)*MO(N)
2180 KT(X)=KT(X)+GE(X,J)
2200 NEXT
2210 NEXT
2220 NEXT
2230 V=I
2240 FORX=1TOV
2250 IFKT(X)<2THENNA$(X)="":KT(X)=0:I=I-
1
2260 NEXT
2280 IFI=0THEN3000
2300 U=0
2310 FORD=2TO8
2320 IFKT(D)=<KT(D-1)THEN2350
2330 K=KT(D):KT(D)=KT(D-1):KT(D-1)=K
2335 N$=NA$(D):NA$(D)=NA$(D-1):NA$(D-1)=
N$
2340 U=1
2350 NEXT
2360 IFU>0THEN2300
2370 GOTO1000
3000 GOSUB5000
3010 PRINT"          SCHOEN
EN DANK FUER DAS NETTE"
3020 PRINT"      SPIELCHEN."
3030 PRINT"          "
3040 PRINT"      BIS ZUM NAECHSTEN MAL ---
"
3050 PRINT"          "
3060 PRINT"                          EU
ER VC-64"
3070 GETA$:IFA$><"X"THENRUN
3080 END
3090 GETA$:IFA$><"C"THEN3090
4000 GOSUB5000
4010 PRINT"";NA$(
X);" STEIGT AUS"
4020 PRINT"--- UND ERHAELT ";KT(X)/
100;"DM!"
4030 PRINT"BEZAHLT? DANN 'C' DRUEC
KEN."
4040 GETA$:IFA$><"C"THEN4040
4050 GOTO1418
5000 PRINT""
5010 FORO=1TO8:PRINT"
                 ":NEXT
5020 RETURN
5100 PRINT"FAITES
 VOTRE JEU."
5110 PRINT"              MESDAMES ET
MESSIEURS!"
5120 RETURN

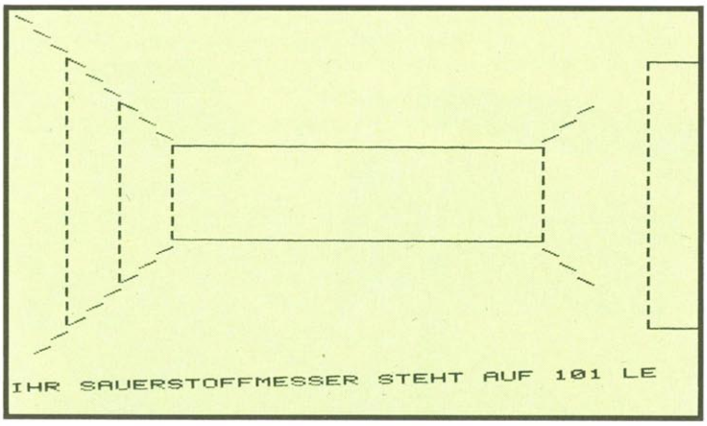
READY.
```

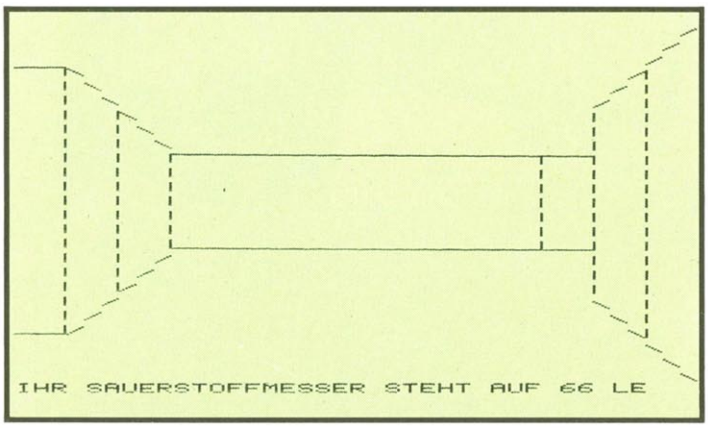
Labyrinth

In der April-Ausgabe von Homecomputer soll eines der Themen die Betrachtung von Strategien zur Erzeugung von Labyrinthen sein.

Hier schon einmal ein kleiner Vorgeschmack. Wie interessant dieses Thema ist, merkt man erst, wenn man versucht, dem Computer das Erzeugen von Irrgärten beizubringen oder, wie alljährlich bei den EURO-MAUS-Wettbewerben zu sehen ist, man Computern das Meistern von Irrgärten zu lehren versucht.

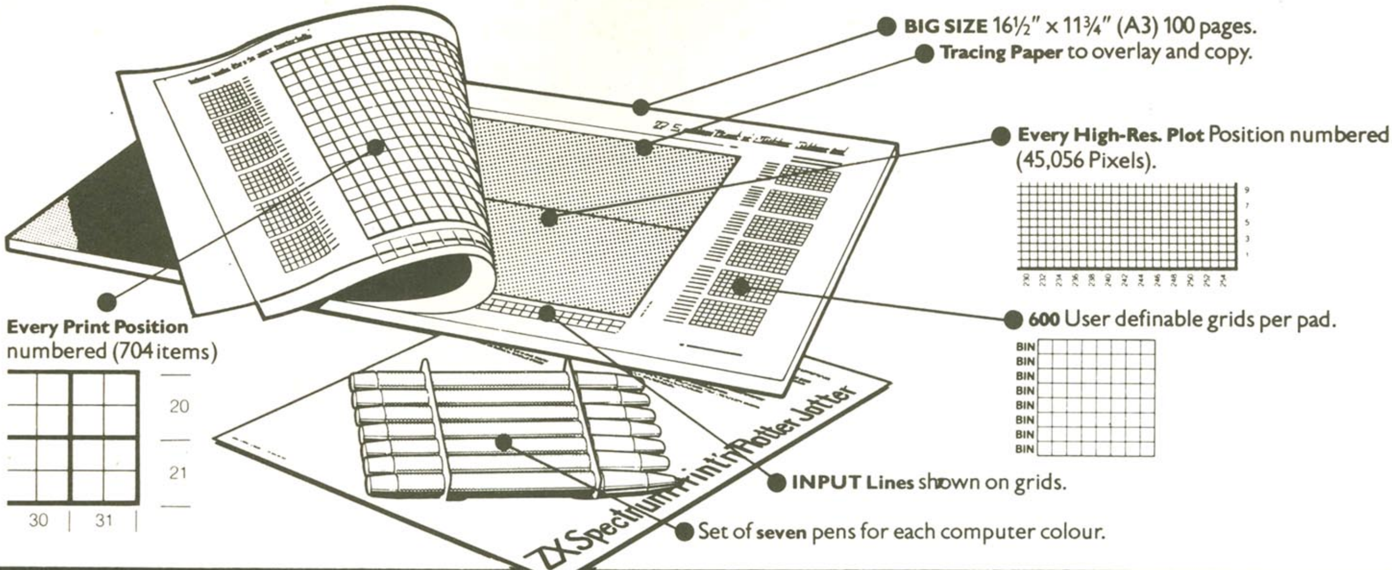
Das vorliegende Programm erzeugt eine 3-D-Ansicht, durch die es schon in einem kleinen Labyrinth schwierig wird, die Orientierung zu gewinnen.






```
3 POKE52,64:POKE54,64:POKE56,64
4 VP=54272:CO=5
5 GOTO900
10 Y=PEEK(TC-U):Z=PEEK(TC-U+LL):IFABS(U)
=1ORDC=0THEN30
20 IFY=32ANDZ=32THENFORA=TC-U+LLTOBC-V-L
LSTEPLL:POKEA,W3:NEXT
30 POKETC,TW:POKEBC,BW:POKETC+VP,CO:POKE
BC+VP,CO
31 B=B+1:IFB<DWTHENTC=TC+U:BC=BC+V:GOTO3
0
40 IFABS(U)=1THENPOKETC,T1:POKEBC,B1:POK
ETC+VP,CO:POKEBC+VP,CO
50 IFTC=BC-LLTHENRETURN
60 FORB=TC+LLTOBC-LLSTEPLL:POKEB,SW:POKE
B+VP,5:NEXT:RETURN
70 F=ML:U=LL+1:V=1-LL:T1=80:B1=122:SW=10
3
80 FORX=4TO6STEP2:TW=77:BW=78:B=0:IFX=4T
HEN100
90 F=MR:U=LL-1:V=-LL-1:SW=101:T1=79:B1=7
6:TW=78:BW=77
100 LM=M(F):LD=W(F)
101 IFF=3ANDOP=ECTHEN140
104 IFLD=0THENLD=2
105 IFQ-LD*INT(Q/LD)=0THENTC=M(X):BC=M(X
+1):GOSUB10:GOTO130
110 TW=99:BW=100:U=U-LL:V=V+LL:N=PEEK(OP
+LM)
120 IFN-D*INT(N/D)=0THENTC=M(X)+DW*LL:BC
=M(X+1)+DW*-LL:GOSUB10
130 IFABS(U)=1THENU=U+LL:V=V-LL
140 M(X)=M(X)+DW*U:M(X+1)=M(X+1)+DW*V:NE
XT
150 IFM=3ANDOP=ECTHENRETURN
160 IFQ-D*INT(Q/D)=0THEN180
170 OP=OP+M(M):Q=PEEK(OP):DC=DC+1:IFDC<C
DTHEN70
175 IFDC=CDTHENRETURN
180 R=PEEK(M(4)-1)
185 IFR=32THENM(4)=M(4)-1:M(5)=M(5)-1:IF
M(4)<>TL+DW*LLTHEN180
190 R=PEEK(M(6)+1)
195 IFR=32THENM(6)=M(6)+1:M(7)=M(7)+1:IF
M(6)<>TR+DW*LLTHEN190
200 FORTC=M(4)TOM(6):POKETC,W1:POKETC+VP
,CO:NEXT
210 FORBC=M(5)TOM(7):POKEBC,W2:POKEBC+VP
,CO:NEXT:RETURN
240 PRINT"IHR SAUERSTOFFMESSER STEHT AUF
"OX"LE";
245 GETA$:IFA$=""THEN245
246 IFA$="P"THENGOSUB5000 :GOTO245
247 PRINT
250 NP=P:Q=PEEK(P):IFA$="G"THENNP=P+M(M)
260 IFNP=PTHEN320
270 IFNP=ECTHENPRINT"AUSGANG ERREICHT!":
GOSUB4000:RUN
285 IFQ-D*INT(Q/D)><0THENP=NP:PRINT"SIE
GEHEN":GOTO380
300 IFRND(1)>.9THENPRINT"IHR RAUMANZUG W
IRD ZERSTOERT";:OL=OL+1
310 PRINT"SIE HABEN EINE WAND GERAMMT":G
OSUB3000:GOTO410
320 NM=M:IFA$="R"THENNM=M+1
330 IFA$="O"THENNM=M+2
340 IFA$="L"THENNM=M+3
350 IFNM=MTHEN240
360 IFNM>3THENNM=NM-4*INT(NM/4)
370 M=NM:PRINT"SIE DREHEN SICH"
380 D=W(M):Q=PEEK(P):OP=P:ML=M-1:MR=M+1
390 IFML<0THENML=3
400 IFMR>3THENMR=0
410 OX=OX-OL:IFOX<0THENPRINT"IHRE LUFT I
ST VERBRAUCHT!":GOSUB4000:RUN
420 T=T+1:IFT=TTTHENTT=1:PRINT"SIE FINDE
N EINE LAMPE":GOSUB3000
430 IFTT=1THENDC=0:GOSUB2000:GOSUB70
435 IFP=MCANDRM=0THENRM=1:GOSUB600
440 M(4)=TL:M(5)=BL:M(6)=TR:M(7)=BR:GOTO
240
500 P=P+1:IFP>EMTHENP=SM
510 A=INT(RND(1)*4):DC=0
520 A=A+1:DC=DC+1:IFDC>3THEN500
530 IFA>3THENA=0
540 M=P+M(A):IFM<SMORM>EMTHEN520
550 CP=PEEK(P):CM=PEEK(M):IFC>0ANDCP=210
THENP=M:GOTO510
560 TM=M-SM:IF(CP=CMORCM<210)ANDC>0THEN5
20
570 ME=TM-L*INT(TM/L):IF(ME=0ANDM(A)=1)O
R(ME=GANDM(A)=-1)THEN520
580 OD=INT(15/W(A)):CP=CP/W(A):POKEP,CP:
CM=CM/OD:POKEM,CM
590 P=M:C=C+1:IFC<HTHEN510
595 RETURN
600 GOSUB2000:PRINT"SIE FINDEN EINE KART
E"
605 A=TL+LL:MS=A:DC=SM:DD=SM+G
610 FORB=DCTODD:FORC=0TO3:CB=PEEK(B):CB=
CB-W(C)*INT(CB/W(C))
620 AA=1:IFC=1ORC=3THENAA=LL
630 IFC>1THENAA=-AA
640 BB=LL/AA:P2=A+AA:P1=P2+BB:P3=P2-BB
645 IFB=PTHENPOKEA,W4:POKEA+VP,CO
650 IFCB=0THENPOKEP1,W3:POKEP2,W3:POKEP3
,W3
651 IFCB=0THENPOKEP1+VP,CO:POKEP2+VP,CO:
POKEP3+VP,CO
660 NEXT:A=A+2:NEXT:DC=DC+L:DD=DD+L:A=MS
+(2*LL):MS=A
670 IFDD<=EMTHEN610
680 GOSUB3000:RETURN
900 GOSUB2000:PRINT"SIE VERSTECKEN SICH
AUF EINEM PLANETEN  IN EINEM LABYRINTH"
910 PRINT"FINDEN SIE DEN AUGANG BEVOR IH
NEN DER    SAUERSTOFF AUSGEHT."
950 TL=1025:BL=1985:TR=TL+38:BR=BL+38:LL
=40:DW=3:CD=6
955 W1=99:W2=100:W3=160:W4=88
960 W(0)=5:W(1)=7:W(2)=3:W(3)=2:L=INT(RN
D(1)*5+6)
970 M(0)=1:M(1)=L:M(2)=-1:M(3)=-L:M(4)=T
L:M(5)=BL:M(6)=TR:M(7)=BR
980 W=INT(RND(1)*5+6):H=L*W-1:G=L-1:C=0:
DC=0:T=0:RM=0
995 SM=16384:EM=SM+H:FORA=SMTOEM:POKEA,2
10:NEXT:P=SM+INT(RND(1)*H)
1000 EC=SM+INT(RND(1)*L):MC=INT(RND(1)*H
)+SM
1010 OX=INT(H/3)+1:IFP<OXTHENOX=-OX
1015 OX=OX+H:OL=1:TT=INT(RND(1)*H/8)+1:G
OSUB500:GOSUB2000
1030 M=0:NP=P:CE=PEEK(EC):CE=CE/W(3):POK
EEC,CE:GOTO380
2000 FORX=0TO25:PRINT:NEXT:RETURN
3000 FORX=0TO5000:NEXT:RETURN
4000 GETA$:IFA$=""THEN4000
4010 RETURN
5000 OPEN4,4:PRINT#4,CHR$(14)
5005 FORI=1024TO2023STEP40:P$=""
5010 FORJ=0TO39:X=PEEK(J+I)
5020 IFX<32THENX=X+64:GOTO5100
5025 IFX<64THEN5100
5030 IFX<96THENX=X+32:GOTO5100
5035 IFX<128THENX=X+64:GOTO5100
5100 P$=P$+CHR$(X)
5150 NEXT:PRINT#4,P$:NEXT
5155 CLOSE4
5160 RETURN

READY.
```



It will improve your ZX SPECTRUM graphics in ways you wouldn't believe possible.

Your new ZX Spectrum is literally packed with sophisticated graphics. Colour. High resolution. Plot. Draw. Circle. Border. Ink. Paper Colours. User-defined characters to name just a few!

That's why we have packed the new Spectrum Print 'n' Plotter Jotter with every facility to exploit your graphics to the full.

After the first few weeks of "playing" with your computer you will want to get down to serious programming and planning in which professional-looking graphics should play a major part.

What better way to work it out than with a Print 'n' Plotter Jotter?

The professional pad

Print 'n' Plotter is not just another programming pad.

Just look at the specifications:

BIG SIZE 16½" x 11¾". 100 Pages — 50 Print Grids and 50 High Resolution Plot Grids.

Printed on high-quality tracing paper, enabling you to overlay the pages for direct co-ordination between PRINT and PLOT or to copy from illustrations, maps, charts, photos etc.

PRINT Grids show all numbered co-ordinates for the 704 screen positions, plus INPUT lines.

PLOT Grids show every numbered co-ordinate for the 45,056 Pixels!

Each pad contains 600 user-definable grids for use with the BIN n, POKE USR "a" function.

And the whole thing is fully bound with fly leaf cover and complete with a set of seven colour pens!

The simple way to get serious

Spectrum Graphics can become very complex, so before you start to program the best way is to work it out on a Print 'n' Plotter and save all those errors!

Take for instance the common CIRCLE. With a Jotter you can establish the exact screen location for the centre in seconds, and it will stop you running out of screen because of a too large radius. Working our DRAW is similar: pre-determine DRAW lines and PLOT positions before you start. With a Jotter you can build-up graphics using every facility with a direct co-ordination between each.

For instance, correct PLOT OVER or PRINT OVER positions will be easy with a Jotter.

See the show for just 95p!

To demonstrate the graphic possibilities with the SPECTRUM JOTTER we have produced a cassette-based Demonstration program for only 95 p (inc VAT and P&P). Why not send for a copy, or order it together with your JOTTER?

Just part of a range of ZX products

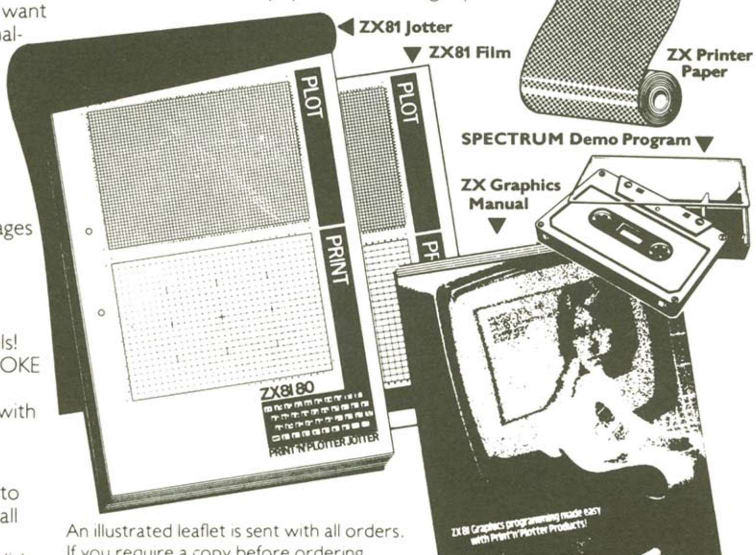
The Spectrum Jotter is, of course, an upgraded version of our popular *ZX81 Print 'n' Plotter Jotter and Film*. For ZX81 owners these are available by direct mail or through a growing number of retailers and compshops.

The ZX81 Jotter is a 100 page Graphics pad that exploits to the full the graphics facilities of that micro. ZX81 Film is a matt film version of the Jotter which is re-usable and ideal for 'copying' graphics.

Our manual: *"ZX Graphics programming made easy"* explains everything you need to know about using the ZX81 products, and when used in conjunction with the Spectrum cassette will prove to be the definitive guide to the subject.

And for ZX users (whether Spectrum or ZX81) we still market Printer Paper at £1 less than Sinclairs!

Why not write and place your order today? Graphics can be a very serious subject . . . Print 'n' Plotter products can make it easier . . . and better in a hundred ways you never thought possible!



An illustrated leaflet is sent with all orders. If you require a copy before ordering write enclosing a first class stamp.

To: Print 'n' Plotter Products (YB) 19 Borough High Street, London SE1 Please forward me the following products:

6th ZX MICROFAIR FEBRUARY 26th NEW HORTICULTURAL HALL LONDON SW1

. . . . . ZX SPECTRUM JOTTERS @ £9.95 each.
. . . . . ZX81 JOTTERS @ £3.50 each.
. . . . . ZX81 PLOTTER FILMS @ £2.25 each.
. . . . . "GRAPHICS PROGRAMMING MADE EASY" MANUALS @ £1.50 ea
. . . . . PACKS OF ZX PRINTER PAPER (5 ROLLS) @ £10.95 per pack.
. . . . . ZX SPECTRUM DEMO CASSETTES @ 95 p each.

PLEASE NOTE ALL PRICES INCLUDE POST, PACKING AND VAT FOR U.K. DELIVERIES (Overseas should add 25% for additional Surface Mail)

Remittance enclosed payable to Print'n'Plotter Products.
Please bill my Access/Barclaycard/Visa/Mastercard No:–

Name: . . . .

Address: . . . .

. . . .

Print'n'Plotter Products©

Most Print'n'Plotter products are available 'over the counter' from the following retailers: Branches of W. H. Smith ● Buffer Micro Shop, Streatham, London ● Denny's Booksellers, London EC1 ● Georges Bookshop, Bristol ● Telford Electronics & Computing, Hollinswood, Telford ● Personal Computer Services, Darwen, Lancs

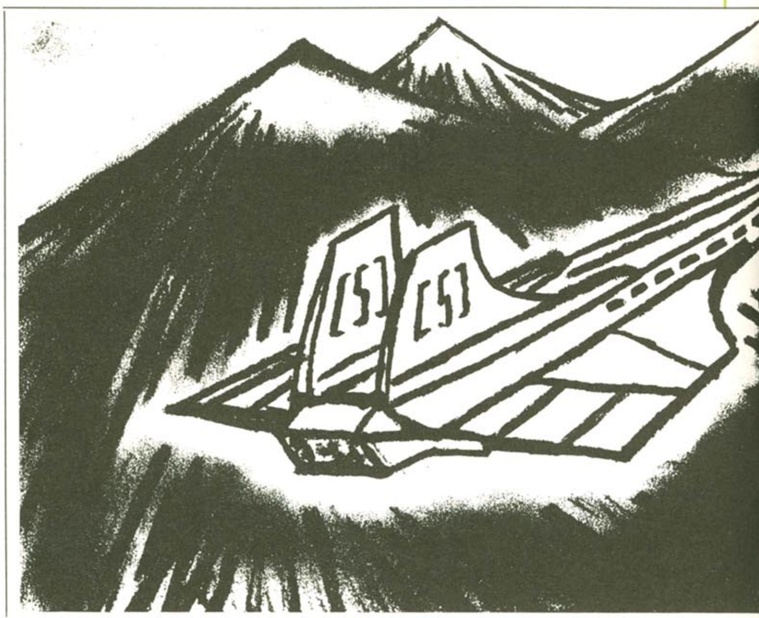
Concorde -

oder Airport! Flug in Gefahr

". . . und bringen Sie den Arzneikoffer mit, ich übernehme . . ."

"... schaffe es aber nicht mehr lange. Auch mir wird auf einmal so schlecht. Fragen Sie nach, ob unter den Passagieren jemand ist, der sich zutraut, das Ding zu landen und machen Sie schnell ..."

Panik an Bord; wenn jetzt nicht einer klaren Kopf behält, stürzt die Maschine mit Sicherheit ins Meer. Übernehmen Sie das Ruder und versuchen Sie, die Maschine zum Flughafen zu fliegen und dort sicher zu landen.



Digital-Uhr & Kalender

Zwei Kurzprogramme zum Thema Zeit

Beide Programme, Uhr und Kalender erklären sich nach dem Starten von selbst. Auf eine genaue Beschreibung kann deshalb verzichtet werden.

Kalender

```
10 PRINT"■■KALENDER":PRINT"■■VON":PRINT"■■RALPH ROESKE":PRINT"■■■DRUECKE ■
> ■"
20 GETA$:IFA$><">"THEN20
30 GOSUB90
40 GETA$:IFA$="A"THEN50
42 IFA$="B"THEN60
44 IFA$="C"THEN70
46 IFA$="D"THEN80
48 GOTO40
50 GOSUB90:PRINT"■■■■*■■■■■■■1.";:GOSUB100:GOSUB200:PRINT"■";W$:S1=S
51 T2=TT:M2=MM:J2=JJ
52 PRINT"■■2.";:GOSUB100:GOSUB200:PRINT"■";W$:S2=S:S3=S-S1:T3=TT
53 M3=MM:J3=JJ:PRINT"■DAZWISCHEN ";:IFS3=1THENPRINT"LIEGT";S3;"TAG":GOTO55
54 PRINT"LIEGEN ";S3;"TAGE"
55 IFS3<0THEN360
56 TT=T2:MM=M2:JJ=J3:GOSUB200:IFS-S1>S3THENJ3=JJ-1:GOTO56
57 GOTO370
58 IFABS(S4)><1ANDS4><0THENPRINT"E"
59 GOTO40
60 GOSUB90:PRINT"■■■■■*■■■■■■■   ";:GOSUB100:GOSUB200:PRINT"■";W$
65 GOTO40
70 GOSUB90:PRINT"■■■■■■■*■■■■■■1.";:GOSUB100:GOSUB200:PRINT"■";W$:S1=S
71 INPUT"■+/- WIEVIEL TAGE";WT
72 WS=WT+S1:JJ=JJ+INT(WT/365.25)
73 TT=1:MM=1:GOSUB200:GOSUB300:IFS>WSTHENJJ=JJ-1:GOTO73
74 TT=31:MM=12:GOSUB200:GOSUB300:IFS<WSTHENJJ=JJ+1:GOTO74
75 TT=1:MM=1
76 GOSUB200:GOSUB300:IFS<WSTHENMM=MM+1:GOTO76
77 MM=MM-1
78 GOSUB200:GOSUB300:IFS<WSTHENTT=TT+1:GOTO78
79 GOTO40
80 GOSUB90:PRINT"■■■■■■■■*■■■■   ";
82 INPUT"      ■      (MONAT.JAHR)■■■■■■■■■■■■■■■■■■■";DA$:DA$="1."+DA$:GOTO400
90 PRINT"■■WAEHLEN SIE:":PRINT"■ A = TAGE ZWISCHEN ZWEI DATEN"
92 PRINT"■ B = WOCHENTAGBESTIMMUNG"
94 PRINT"■ C = DATUM NACH ANZAHL VON TAGEN"
96 PRINT"■ D = DRUCKE KALENDER"
98 RETURN
100 INPUT"DATUM:■ (TAG.MONAT.JAHR)■■■■■■■■■■■■■■■■■■■";DA$
110 F=0:FORI=1TOLEN(DA$):IFMID$(DA$,I,1)="."ORMID$(DA$,I,1)=" "THENF=F+1:P(F)=I
115 NEXT
120 IFF><2THENPRINT"■FALSCHE EINGABE !";:GOTO122
121 GOTO130
122 FORI=1TO500:NEXT:PRINT"■■■■■■■■■■■■■■■■■■■                    "
123 PRINT"■■                                      ":PRINT"■■■■";
124 GOTO100
130 TT=VAL(LEFT$(DA$,P(1)-1)):MM$=MID$(DA$,P(1)+1,P(2)-P(1)):MM=VAL(MM$)
135 JJ=VAL(MID$(DA$,P(2)+1))
140 RESTORE:GOSUB190
150 IFJJ>1582THEN180
160 IFJJ=1582ANDMM>10THEN180
170 IFJJ=1582ANDMM=10ANDTT>14THEN180
175 F=0:GOTO120
180 RETURN
190 FORI=1TO12:READM$:IFLEFT$(MM$,3)=LEFT$(M$,3)THENMM=I
195 NEXT:IFMM=0THEN175
196 RETURN
198 DATAJANUAR,FEBRUAR,MAERZ,APRIL,MAI,JUNI
199 DATAJULI,AUGUST,SEPTEMBER,OKTOBER,NOVEMBER,DEZEMBER
200 Z=0:IFMM<3THEN260
210 IFJJ/4><INT(JJ/4)THEN250
220 IFJJ/100><INT(JJ/100)THEN240
230 IFJJ/400><INT(JJ/400)THEN250
240 Z=1
250 V=INT((306*MM-324)/10):GOTO270
260 V=(MM-1)*31
270 Z=Z+(JJ-1)*365+INT((JJ-1)/4):Z=Z-INT((JJ-1)/100)+INT((JJ-1)/400)
280 S=Z+V+TT:W=S-INT(S/7)*7
290 POKE62,206:POKE63,10:FORI=0TOW:READW$:NEXT
295 RETURN
298 DATA"SONNTAG    ","MONTAG     ","DIENSTAG   ","MITTWOCH   "
299 DATA"DONNERSTAG ","FREITAG    ","SAMSTAG    "
300 T1$=STR$(TT):M1$=STR$(MM):J1$=STR$(JJ)
310 GD$=MID$(T1$,2)+"."+MID$(M1$,2)+"."+MID$(J1$,2)
320 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■????  = ";W$;" ";GD$;"       ":RETURN
350 RESTORE:FORI=1TO12:READM$:NEXT:PRINTPEEK(62);PEEK(63):LIST290
360 WV=T2:T2=T3:T3=WV:S3=ABS(S3):WV=S1:S1=S2:S2=WV:WV=M2:M2=M3:M3=WV
361 WV=J2:J2=J3:J3=WV:GOTO56
370 S4=S2-S:J4=J3-J2
371 IFABS(J4)><1THENPRINT"■ODER";J4;"JAHRE";:GOTO373
372 PRINT"■ODER";J4;"JAHR";
373 IFS4><0THENPRINT" &";S4;"TAG";
375 GOTO58
400 GOSUB110:T1=TT:M1=MM:J1=JJ
410 GOSUB200:S1=S:W1=W:MM=MM+1:IFMM=13THENMM=1:JJ=JJ+1
420 GOSUB200:S2=S
425 PRINT"■■■■■SO■■■MO■■■DI■■■MI■■■DO■■■FR■■■SA"
426 RESTORE:FORI=1TOM1:READM$:NEXT:PRINT"■■";TAB(16-(LEN(M$)+5)/2);M$;J1
430 PRINT"■■■■■■■■■";:FORI=0TOW1:PRINT"■■";:NEXT
440 FORI=1TOS2-S1:PRINTRIGHT$(STR$(I),2);
445 W1=W1+1:IFW1=7THENW1=0:PRINT"■■■■■■■■■■■■■■■";:GOTO460
450 PRINT"■■■";
460 NEXT
470 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■ ↑ ■ = NAECHSTER MONAT ■"
475 PRINT"■    ■ = ZURUECK ZUM START"
480 GETA$:IFA$="↑"THEN490
485 IFA$><" "THEN480
487 GOTO30
490 TT=1:MM=M1+1:IFMM=13THENMM=1:JJ=J1+1
495 M1=MM:J1=JJ:GOTO410
```

Digital-Uhr

```
1 POKE59490,255
5 INPUT"■WIEVIEL UHR IST ES";TI$
10 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■•■■■•■■■■■■■■■■■■■■■■■•■■■•"
11 PRINT"■■■■■■■■■■■■■■■■■                                   ■"
12 PRINT"■■■                              ■"
20 T1=VAL(LEFT$(TI$,1)):PRINT"■■■■■■";:IFT1=0THENT1=10
25 ONT1GOSUB110,120,130,140,150,160,170,180,190,200
30 T2=VAL(MID$(TI$,2,1)):PRINT"■■■■■■■■■■■■";:IFT2=0THENT2=10
35 ONT2GOSUB110,120,130,140,150,160,170,180,190,200
40 T3=VAL(MID$(TI$,3,1)):PRINT"■■■■■■■■■■■■■■■■■■■■■■■■";:IFT3=0THENT3=10
45 ONT3GOSUB110,120,130,140,150,160,170,180,190,200
50 T4=VAL(MID$(TI$,4,1)):PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■";:IFT4=0THENT4=10
55 ONT4GOSUB110,120,130,140,150,160,170,180,190,200
60 T5=VAL(MID$(TI$,5,1)):PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■";:IFT5=0THE
NT5=10
65 ONT5GOSUB110,120,130,140,150,160,170,180,190,200
70 T6=VAL(RIGHT$(TI$,1)):PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■";:IFT
6=0THENT6=10
75 ONT6GOSUB110,120,130,140,150,160,170,180,190,200
80 S2=VAL(MID$(TI$,2,1)):S4=VAL(MID$(TI$,4,1))
82 IFS2><T2THEN20
84 IFS4><T4THEN40
86 IFT6=10THEN60
90 GOTO70
110 PRINT"   •■■■■■  ••■■■■■ • •■■■■■•  •■■■■■   •■■■■■   •■■■■■   •■"
115 RETURN
120 PRINT" •• ■■■■■•  •■■■■■   •■■■■■  • ■■■■■ •  ■■■■■•   ■■■■■•••■"
125 RETURN
130 PRINT" •• ■■■■■•  •■■■■■   •■■■■■  • ■■■■■   •■■■■■•  •■■■■■ •• ■"
135 RETURN
140 PRINT"•   ■■■■■•   ■■■■■•   ■■■■■• • ■■■■■•••■■■■■  • ■■■■■  • ■"
145 RETURN
150 PRINT"••••■■■■■•   ■■■■■•   ■■■■■••• ■■■■■   •■■■■■•  •■■■■■ •• ■"
155 RETURN
160 PRINT" •• ■■■■■•  •■■■■■•   ■■■■■••• ■■■■■•  •■■■■■•  •■■■■■ •• ■"
165 RETURN
170 PRINT"••••■■■■■   •■■■■■   •■■■■■  • ■■■■■ •  ■■■■■•   ■■■■■•   ■"
175 RETURN
180 PRINT" •• ■■■■■•  •■■■■■•  •■■■■■ •• ■■■■■•  •■■■■■•  •■■■■■ •• ■"
185 RETURN
190 PRINT" •• ■■■■■•  •■■■■■•  •■■■■■ •••■■■■■   •■■■■■•  •■■■■■ •• ■"
195 RETURN
200 PRINT" •• ■■■■■•  •■■■■■•  •■■■■■•  •■■■■■•  •■■■■■•  •■■■■■ •• ■"
205 RETURN
```

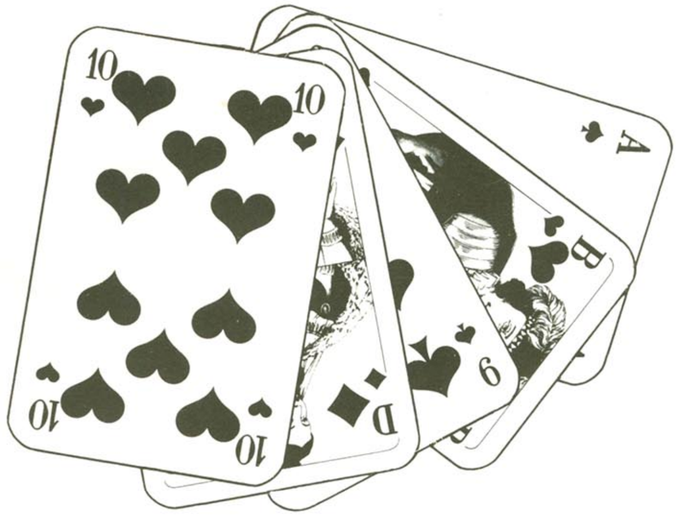


Zum Programm Kalender ist aber zu bemerken, daß nicht nur ein Kalenderblatt auf den Bildschirm gedruckt wird, sondern auch das Datum vor oder nach einer bestimmten Anzahl von Tagen ermittelt werden kann.
Außerdem sind die üblichen Abfallprodukte eines Kalenderprogramms zu erfragen, nämlich die Wochentagsbestimmung für ein beliebiges Datum, sowie die Anzahl der Tage zwischen zwei Daten.
Allerdings ist nach Eintasten des Programm oder nach Änderungen vor der Zeile 300 mit RUN 350 ein Tuning der DATA-Pokes in Zeile 290 vorzunehmen.
Die beiden ausgedruckten Werte werden in die gelistete Zeile 290 übertragen. Es handelt sich um die Zeropage-Bytes, die den Zeiger auf die aktuelle DATA-Adresse bilden.
Diese Möglichkeit, den Zeiger auf das als nächstes zu lesende Datawort einzustellen, eignet sich natürlich besonders in langen Programmen mit vielen Datazeilen.
Es kann dann noch besser die Methode angewandt werden, die beim VC-Zeichengenerator benutzt wird, nämlich: entsprechende Zeile per Programm suchen und die Werte hineinpoken.

```
 10 REM  ***  CONCORDE  ***
 20 REM
 30 PRINT"□";
 50 PRINT"■****************************************"
 60 PRINT"■■**";TAB(14);"CONCORDE"
 65 PRINT"■■**"
 70 PRINT"■■****************************************"
 100 INPUT"BENOETIGEN SIE DIE SPIELREGELN";A$
 120 IFLEFT$(A$,1)="N"THEN250
 150 PRINT"SIE FLIEGEN EIN FLUGZEUG 2000 MEILEN UNDVERSUCHEN ZU LANDEN."
 160 PRINT"NAVIGATION:"
 170 PRINT"DER COMPUTER FRAGT 'LINKE, RECHTE TRAG- FLAECHE (GRAD)'"
 180 PRINT"SIE SCHREIBEN WIEVIEL GRAD DIE WINKEL   DER TRAGFLAECHEN BETRAGEN SOL
LEN."
 190 PRINT"DANN FRAGT DER COMPUTER NACH DER BE-    SCHLEUNIGUNG.";
 200 PRINT"UND SIE TASTEN EIN, WIEVIELSCHNELLER SIE FLIEGEN WOLLEN.";
 210 PRINT"EINE NEGA- TIVE ZAHL, WENN SIE LANGSAMER WERDEN    WOLLEN."
 220 PRINT"SIE ERHALTEN EINE BESTIMMTE MENGE SPRIT.IST DER SPRIT ALLE, GLEITEN S
IE!"
 240 PRINT"VIEL GLUECK!!!"
 250 INPUT"■SIND SIE 'NEULING', 'ERFAHRENER', 'EX-  PERTE' OR 'SUPER EXPERTE'■";
A$
 255 PRINT"□"
 260 IFLEFT$(A$,1)="N"THENS=61:GOTO400
 270 IFMID$(A$,2,1)="R"THENS=41:GOTO400
 280 IFMID$(A$,2,1)="X"THENS=21:GOTO400
 290 IFLEFT$(A$,1)="S"THENS=11:GOTO400
 300 PRINT"NON KAPITO":GOTO250
 400 Z=2000:Q=0:Y=600:X=INT(RND(1)*30)-15:W=0:V=400
 410 Z9=Z:Y9=Y:X9=X*15:W9=W:V9=V:S9=S
 460 IFZ<-100GOTO1030
 470 IFV<0GOTO1060
 480 IFY<150GOTO1090
 490 IFV=0GOTO860
 500 S=S-Y/600:X=X+W
 530 PRINT"□FLUGHOEHE (FEET)     :";INT(V*100)/100;TAB(30);"("V9")"
 532 PRINT"GESCHWINDIGKEIT      :";INT(Y*100)/100;TAB(30);"("Y9")"
 534 PRINT"POSITION (FEET)      :";INT(X*1500)/100;TAB(30);"("X9")"
 536 PRINT"WINKEL (GRAD)        :";INT(W*100)/100;TAB(30);"("W9")"
 538 PRINT"ENTF. ZUR ROLLBAHN   :";INT(Z*100)/100;TAB(30);"("Z9")"
 540 PRINT"SPRIT                :";INT(S*100)/100;TAB(30);"("S9")"
 550 V9=INT(V):Y9=INT(Y):X9=INT(X*15):W9=INT(W):Z9=INT(Z):S9=INT(S)
 560 GOTO580
 570 PRINT"NUR -45 BIS 45 SIND MOEGLICH!"
 580 INPUT"■LINKE, RECHTE TRAGFL. (GRAD)";L,R
 600 L=L-L*2:R=R-R*2
 620 IFL<-45ORL>45ORR<-45ORR>45THEN570
 660 T=L-R:W=W+T:N=L+R:A=N/2:V=V+A
 710 IFS>0THEN790
 720 IFQ<>0GOTO740
 730 PRINT"YOU'RE OUT OF GAS!"
 740 Q=1:A=A*1.1:A=A-A*2:A=A-5:GOTO830
 790 INPUT"■BESCHLEUNIGUNG (-100 TO 100)";A
 810 IFA<-100ORA>100THEN790
 830 Y=Y+A:Z=Z-Y/8:GOTO460
 860 IFX<-5ORX>5GOTO1030
 880 PRINT"SIE SIND AUF DER ROLLBAHN."
 890 D=Y
 900 D=INT(D):PRINT"GESCHWINDIGKEIT";D
 920 PRINT"ENTFERNUNG ZUM ROLLBAHNENDE:";100-ABS(Z):GOTO950
 940 PRINT"NUR 30 M.P.H. SIND JE ABBREMSUNG MOEGL.!"
 950 INPUT"ABBREMSUNG";P
 970 IFP>30GOTO940
 980 D=D-P:Z=Z-D/8
 1000 IFZ<-100THEN1030
 1010 IFD=0THEN1110
 1020 GOTO900
 1030 PRINT"■ROLLBAHN VERFEHLT.■■":GOTO250
 1060 PRINT"■CRASH-LANDUNG!■■":GOTO250
 1090 PRINT"■ABGESTUERZT! SIE KOENNEN NICHT SO LANG- SAM FLIEGEN!■■":GOTO250
 1110 PRINT"■PERFEKTE LANDUNG!■■":GOTO250
 1130 END
READY.
```

51

Können Sie sich noch erinnern? Es muß in der 4. oder 5. Klasse gewesen sein. Damals kannte man das Wort "Stress in der Schule" noch nicht. Freistunden waren Spielstunden und in den Pausen trumpfte man mit Spielkarten, statt mit Fäusten auf. Gute alte Zeit...





Aus jenen Tagen stammt offensichtlich meine Liebe zu den netten Kartenspielchen wie Mau-Mau, Siebzehn-und-Vier und natürlich 51.
Für das Programmieren von Computerspielen sind solche Glücksspiele besonders gut geeignet, wie man aus den unzähligen Spielen, die aus dieser Kategorie auf dem Markt sind, sieht. Gibt es dabei sogar noch eine rationelle Strategie, die man in das Programm einbauen kann, dann lohnt es sich um so mehr, dieses Spiel gegen einen Computer zu versuchen.





Bei 51 gibt es solch eine, die jeder selbst leicht entdecken wird. Folgen beide Partner dieser gleichen Strategie, so ist das Spiel wieder zum reinen Glücksspiel geworden, bei dem es nur noch darauf ankommt, wer die besseren Karten erwischt hat.
Freilich könnte man ein, in Glücksspielen unschlagbares Computerprogramm schreiben, indem man einfach dem Computer 'ermöglicht', in die Karten des Spielers zu schielen. Das bringt dem Programmierer zwar Anerkennung (sofern er nicht entlarvt wird), ist aber wenig befriedigend. Das abgedruckte Programm meidet solche faulen Tricks und ist trotzdem ein gleichwertiger Gegner.
Die Spielregeln sind in den Zeilen 135 bis 165 zu finden.
Viel Spaß

```
5 DIMC$(32),C(32)
10 DATA" A",1," K",4," D",3," B",2,"10",-10," 9",0," 8",8," 7",7
20 FORI=1TO8:C$(I)=" ♣":NEXT
25 FORI=9TO16:C$(I)=" ♠":NEXT
30 FORI=17TO24:C$(I)=" ♥":NEXT
35 FORI=25TO32:C$(I)=" ♦":NEXT
40 FORJ=0TO3:FORI=1TO8:READD$:C$(I+J*8)=C$(I+J*8)+D$+" ":READC(I+J*8):NEXT
45 RESTORE:NEXT
100 PRINT"♥":GOSUB180
105 PRINT"VON RRALPH ROESKE■"
110 PRINT
115 PRINT:PRINT:PRINT
120 INPUT"KENNEN SIE DIE SPIELREGELN";JN$:IFLEFT$(JN$,1)="J"THEN200
125 IFLEFT$(JN$,1)<>"N"THEN100
130 PRINT"♥":GOSUB180:PRINT:PRINT"RSPIELREGELN:"
135 PRINT:PRINT"BEIM ABLEGEN DER HANDKARTEN IST DARAUF"
140 PRINT"ZU ACHTEN, DASS DER GESAMTWERT DER KAR-"
145 PRINT"TEN DES TALONS NICHT UEBER 50 ZAEHLT -"
150 PRINT"WER ZUERST 51 ODER MEHR ERREICHT, HAT   VERLOREN!"
155 PRINT:PRINT"DIE KARTEN ZAEHLEN WIE FOLGT:"
160 PRINT:PRINT"A =  1  *  K =  4  *  D =  3  *  B =  2"
165 PRINT:PRINT"10=-10  *  9 =  0  *  8 =  8  *  7 =  7"
170 PRINT:PRINT:INPUT"ALLES KLAR";JN$:GOTO200
180 PRINT"SR##########################################                  ";
182 PRINT"                   ##                  ' 5 1 '        ";
184 PRINT"          ##                                        ##################";
186 PRINT"###########################■"
188 RETURN
200 PRINT"♥":GOSUB180:PRINT:PRINT"WER DIE HOEHERE KARTE ZIEHT BEGINNT.":PRINT
205 B1=INT(RND(1)*32+1):PRINT"ZIEHEN SIE EINE KARTE (=-TASTE DRUECKEN)"
210 GETA$:IFA$<>"="THEN210
215 B2=INT(RND(1)*32+1):IFB2=B1THEN215
220 A1=B1:A2=B2
230 IFA1>8THENA1=A1-8:GOTO230
235 IFA2>8THENA2=A2-8:GOTO235
240 F1=0:IFA2<A1THENF1=1
245 IFA2=A1THENIFB2<B1THENF1=1
250 PRINT:PRINT:PRINT"     IHRE KARTE        MEINE KARTE"
255 PRINT:PRINT"     R     ■           R     ■"
260 PRINT"     R";C$(B2);"■           R";C$(B1)
265 PRINT"     R     ■           R     ■"
270 IFF1=1THENPRINT:PRINT"SIE FANGEN AN."
275 IFF1=0THENPRINT:PRINT"ICH FANGE AN."
280 T=TI
285 IFTI-T<300THEN285
300 REM    MISCHEN DER KARTEN
301 PRINT"♥":GOSUB180:Z=0
302 M1=INT(RND(1)*32+1):M2=INT(RND(1)*32+1):Z=Z+1
305 C$=C$(M1):C$(M1)=C$(M2):C$(M2)=C$:C=C(M1):C(M1)=C(M2):C(M2)=C
310 PRINT"SQQQQQQQQQQR     ■  R     ■SQQQQQQQQQQQQQR     ■  R     ■"
315 PRINT"SQQQQQQQQQQQR";C$(M1);"■  R";C$(M2);"■   WIR MISCHEN DIE KARTEN."
320 IFZ<32THEN302
325 PRINT"SQQQQQQQQQQQQQQQQENDE DES MISCHENS: LEERTASTE DRUECKEN."
330 GETA$:IFA$<>" "THEN302
400 REM  KARTEN VERTEILEN
405 AZ=10
410 FORI=1TO5:CC$(I)=C$(I):CC(I)=C(I):NEXT
415 FORI=1TO5:SC$(I)=C$(I+5):SC(I)=C(I+5):NEXT
420 PRINT"♥"
500 REM    SPIELER SPIELT AUS
501 IFF1=0ANDAZ=10THENGOSUB180:GOTO600
505 GOSUB180:AZ=AZ+1
510 PRINT"SQQQQQQQQQ R     ■  R     ■  R     ■  R     ■  R     ■"
515 PRINT"  R";SC$(1);"■  R";SC$(2);"■  R";SC$(3);"■  R";SC$(4);"■  R";SC$(5
)
520 PRINT" R     ■  R     ■  R     ■  R     ■  R     ■"
525 PRINT:PRINT"  '1'     '2'     '3'     '4'     '5'"
530 PRINT:INPUT"WELCHE KARTE SPIELEN SIE AUS";A
535 IFA<1ORA>5THENPRINT"IIII":GOTO530
540 IFSC$(A)="     "THENPRINT"IIII":GOTO530
545 TL=TL+SC(A)
550 TL$=SC$(A):SC$(A)="":SC(A)=0:IFAZ<33THENSC$(A)=C$(AZ):SC(A)=C(AZ)
555 IFAZ>32THENSC$(A)="     "
560 IFA=1THENPRINT"SQQQQQQQQQQ R     ■"
562 IFA=2THENPRINT"SQQQQQQQQQQ]]]]]]]]]]]R     ■"
564 IFA=3THENPRINT"SQQQQQQQQQQ]]]]]]]]]]]]]]]]]]]]]R     ■"
566 IFA=4THENPRINT"SQQQQQQQQQQ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]R     ■"
568 IFA=5THENPRINT"SQQQQQQQQQQ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]R     ■"
570 PRINT"SQQQQQQQQQQQQQQQQQQ]]]]]]]]]]]]]]]]]]]]R     ■"
575 PRINT"         TALON     R";TL$;"■    ";TL:PRINT"               R     ■"
580 IFTL>50THENPRINT:PRINT"SIE HABEN VERLOREN!":GOTO800
582 IFF1=1ANDAZ=>41THENPRINT:PRINT:PRINT"UNENTSCHIEDEN":GOTO800
585 T=TI
590 IFTI-T<180THEN590
600 REM  PET SPIELT AUS
605 FORI=2TO5:IFCC(I)>CC(I-1)THEN690
610 NEXT:AZ=AZ+1
611 IFCC(1)+TL<51ANDCC(1)><-20THENTL$=CC$(1):TL=TL+CC(1):L=1:GOTO620
612 IFCC(2)+TL<51ANDCC(2)><-20THENTL$=CC$(2):TL=TL+CC(2):L=2:GOTO620
613 IFCC(3)+TL<51ANDCC(3)><-20THENTL$=CC$(3):TL=TL+CC(3):L=3:GOTO620
614 IFCC(4)+TL<51ANDCC(4)><-20THENTL$=CC$(4):TL=TL+CC(4):L=4:GOTO620
615 IFCC(5)+TL<51ANDCC(5)><-20THENTL$=CC$(5):TL=TL+CC(5):L=5:GOTO620
616 FORI=5TO1STEP-1:IFCC(I)><-20THENTL$=CC$(I):TL=TL+CC(I):GOTO620
617 NEXT
620 IFAZ<33THENCC$(L)=C$(AZ):CC(L)=C(AZ)
625 IFAZ>32THENCC$(L)="R     ■":CC(L)=-20
630 PRINT"SQQQQQQQQQQQQQQQQ                                    "
632 PRINT"                R     ■"
635 PRINT"]]]]]]]]]]TALON    R";TL$;"■    ";TL
640 PRINT"]]]]]]]]]]]]]]]]]]]]R     ■"
645 IFTL>50THENPRINT:PRINT"ICH HABE VERLOREN!":GOTO800
647 IFF1=0ANDAZ=>41THENPRINT:PRINT:PRINT"UNENTSCHIEDEN":GOTO800
650 GOTO500
690 CC=CC(I):CC(I)=CC(I-1):CC(I-1)=CC:CC$=CC$(I):CC$(I)=CC$(I-1):CC$(I-1)=CC$
695 GOTO605
800 IFF1=0THENF1=1:GOTO810
805 IFF1=1THENF1=0
810 PRINT:INPUT"NOCHMAL";JN$:IFLEFT$(JN$,1)="J"THENPRINT"♥":GOSUB180:TL=0:GOTO2
70
```

Submarine-Hunt

Ein Spielprogramm für den Sharp MZ-80 A

Grundlage des Programms sind die beiden, mit zweidimensionalen Tabellen unterlegten Spielfelder.
In der Aufbauphase werden diese Tabellen durch Zweifachindiziierung (X für horizontal, Y für vertikal) mittels Zufallszahlen aufgerufen und mit "1" gesetzt. INPUT "Anzahl";N in Zeile 113 bestimmt die Anzahl der Durchläufe einer FOR-NEXT-Schleife für die Menge der zu belegenden Felder. Vor dem Laden der Tabellenplätze erfolgt eine Überprüfung des Inhalts, um Doppelbelegungen zu vermeiden.

In der Spielphase werden nun, durch Eingabe von Horizontal- und Vertikalkoordinaten, die jeweiligen Felder auf ihren Inhalt überprüft. Ist dieser "1", so bedeutet das einen Treffer.
Über eine Cursor-Vorwahl erhält das betreffende Feld das Graphikzeichen "X". Bei Tabelleninhalt "0" wird das entsprechende Quadrat mit " " belegt. Beide möglichen Ergebnisse sind akustisch unterlegt.

In der Folge wechselt das Spielrecht an den Gegner; im Wechsel, bis einer der Spielpartner die vorgewählte Anzahl von U-Booten erreicht hat. Nach anschliessender Anzeige von Gewinner und Quote wird, auf Anforderung ein neuer Durchgang vorbereitet.

H.-J. Friedemann



Spielprogramm für MZ-80A

xx U-Boot-Jagd xx

```
   4 GOTO 9ØØ
   5 ? "C"
   6 CLR
  10 ?:?:?:?
  20 ?
  30 ?
  40 ?"       Ø123456789Ø123456789"
  50 FOR P=ØTO9
  60 ?"
  70 NEXT P
  80 ?"
  90 ?"              A             B"
 100 DIM A(9,9)
 110 DIM B(9,9)
 111 ?
 112 ?"WIEVIELE U-BOOTE SOLL ICH VERSTECKEN?"
 113 INPUT "ANZAHL";N
 114 IF N>1ØØ GOTO 5
 115 IF N<=Ø GOTO 5
 120 FOR M=1 TO N
 130 X=INT(1Ø×RND(1))
 140 Y=INT(1Ø×RND(1))
 150 I=A(X,Y)
 160 IF I=1 GOTO13Ø
 170 A(X,Y)=1
 180 X=INT(1Ø×RND(1))
 190 Y=INT(1Ø×RND(1))
 200 I=B(X,Y)
 210 IF I=1 GOTO 18Ø
 220 B(X,Y)=1
 230 NEXT M
 240 CURSOR Ø,Ø:PRINT"FERTIG!  xxWER WILL BEGINNEN?xx"
 250 ?"                 (A=1  B=2)"
 251 INPUT C
 260 ON C GOTO 3ØØ,5ØØ
 300 CURSOR Ø,Ø:?"                                  "
 305 ?"                                             "
 306 ?"                                             "
 308 CURSOR Ø,Ø
 310 ?"    SPIELER A! KOORDINATEN EINGEBEN"
 330 INPUT "HORIZ.";X
 331 INPUT "VERT.";Y
 332 IF X>9 GOTO 5ØØ
 333 IF Y>9 GOTO 5ØØ
 340 I=B(X,Y)
 350 IF I=1THEN GOTO 4ØØ
 360 IF I=ØTHEN CURSOR 2Ø+X,Y+8
 361 ?"■"
 362 MUSIC"A"
 380 GOTO 5ØØ
 400 MUSIC"+A"
 405 B(X,Y)=Ø
 410 CURSOR 2Ø+X,Y+8
 420 ?"X"
 430 A1=A1+1
 440 CURSOR Ø,1Ø:?"A:";A1
 445 IF A1=N GOTO 8ØØ
 500 CURSOR Ø,Ø:?"                         "
 501 ?"                                    "
 502 ?"                                    "
 503 ?"                                    "
 504 CURSOR Ø,Ø
```



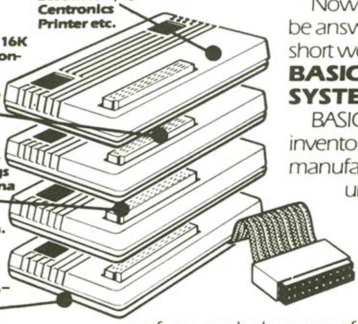
```
 510 ?"    SPIELER B! KOORDINATEN EINGEBEN"
 520 INPUT"HORIZ.";X
 530 INPUT"VERT.";Y
 531 IF X>9 GOTO 3ØØ
 532 IF Y>9 GOTO 3ØØ
 540 I=A(X,Y)
 550 IF I=1 THEN GOTO 7ØØ
 560 IF I=Ø THEN CURSOR 9+X,Y+8
 561 ?"■"
 562 MUSIC"A"
 570 GOTO 3ØØ
 700 MUSIC"+A"
 705 A(X,Y)=Ø
 710 CURSOR 9+X,Y+8
 720 ?"X"
 725 A2=A2+1
 726 CURSOR Ø,15:?"B:";A2
 728 IF A2=N GOTO 8ØØ
 730 GOTO 3ØØ
 800 ?"C"
 810 IF A1>A2 THEN ?"     xx A GEWINNT MIT";A1;":";A2;"PUNKTEN xx"
 820 IF A2>A1 THEN ?"     xx B GEWINNT MIT";A2;":";A1;"PUNKTEN xx"
 830 ?:?:?
 840 ?"              ?? NEUES SPIEL??"
 845 ?:?:?
 850 INPUT"            JA=1,NEIN=0";Q
 860 IF Q=1 GOTO 5
 870 END
 900 ?"C":?
 910 ?"            xxU-BOOT-JAGDxx"
 920 ?
 930 ?"Ihr Spielfeld umfaßt 2 Felder mit"
 935 ?"1Ø x 1Ø Kaestchen."
 950 ?"Ich verteile die von Ihnen gewuenschte Anzahl von U-Booten
      (1-1ØØ)"
 960 ?"mittels Zufallzahlen!"
 965 ?"Ihre Aufgabe ist es,"
 970 ?"durch Eingabe von Horizontal-u."
 975 ?" Vertikalkoordinaten,"
 980 ?"das Feld zu bestimmen, auf dem Sie       das U-Boot vermuten."
 985 ?:?
 990 ?"BEI EINER EINGABE > 9 GEHT DAS"
 991 ?"SPIELRECHT AUTOMATISCH AN DEN"
 992 ?"        GEGENSPIELER!!!"
 993 ?
1000 ?"Wenn Sie alles gelesen haben,"
1100 ?"geben Sie bitte eine 1 als Zeichen ein."
1110 ?
1200 ?"            xxVIEL ERFOLGxx"
1210 ?:?
1300 INPUT"FERTIG?";S
1400 IF S=1 GOTO 5
1500 END
```

HOW TO GROW UP TO A REAL COMPUTER



If you're enthusiastic about microcomputing, sooner or later you'll ask yourself the question . . . "where do I go from here?"

How BASICARE build into a complete computing package.



This is particularly true if you own a micro with limited expansion and hardware peripheral options . . . like the ZX81.

Now your question can be answered in three short words:

BASICARE MICRO SYSTEMS

BASICARE are the inventors, developers and manufacturers of a totally unique microcomputing upgrade system.

It's the sort of system others have dreamed about . . . a series of separate modules that perform a whole range of microcomputing functions that simply (and firmly) stack together.

If you're confused . . . don't be . . . you only have to think of it in terms of the way Hi-FI equipment has evolved.

You buy the hardware you want and add to the system! Each module may have a separate function or integrate functions. And when you want more . . . you add more!

In short you can develop a whole range of hardware options that fit together to form a complete package . . . "Computing" in the real sense of the word!

And what a package!

Apart from it's good looks and stability under working conditions, BASICARE MICRO SYSTEMS offers a fantastic range of micro options for ZX81 users.

Of course, such a system needs a starting point from which to expand. The heart of BASICARE'S system is a unique computer interface which we call PERSONA.

This one unit simply plugs into your ZX81 without modification and acts as the "brains" of the whole operation.

Thereafter you choose how you want to expand your micro by simply plugging-in more modules.

Look at the choice you have:

PERSONA – An interface module to enable an ORGANIC MICRO to grow on the ZX81.
MINIMAP – A memory mapping device to extend the address space of the ZX81 from 64K bytes to 1 M bytes.
RAM 08 – A low cost, low power memory expandable from 2K to 8K.
RAM 16 – 16K Add on memory at remarkably low cost.
RAM 64 – A TRUE 64K Add on memory.
DROM – Ultra low power memory backed by rechargeable battery for non-volatile storage of programs and data.
TOOLKIT – A module fully socketed to take up to 8K bytes of utilities in EPROM/ROM.
PERICON a – A general purpose, user programmable device providing 24 lines of input/output.
PERICON b – 24 lines of heavy duty output to access and control the outside world.
PERICON c – A module to drive 80 column printer with Centronics type parallel interface.
USERFONT option – User definable characters available for RAM 08, DROM and TOOLKIT.

Of course, there's lots more BASICARE Modules under development including joysticks, EPROM Programmer, Floppy Disc Controller and much more.

BASICARE is the sort of system that GROWS when you grow . . . and remember, when you change your micro, you will be able to change to a PERSONA unit to suit your new equipment!

In short, BASICARE will serve you forever . . . no matter how big you want to grow!

"As space is limited to describe BASICARE products, we have produced a fully documented brochure . . . send for it today (enclosing stamp please).



BASICARE MICRO SYSTEMS are available by mail simply by sending today together with cheque/P.O./Access or Barclaycard No. Please indicate clearly your exact requirements.

Name ______________________

Address ______________________

____

PERSONA	@ **£30.25**	DROM(2K)	@ **£39.50**
MINIMAP	@ **£35.95**	TOOLKIT	@ **£22.20**
RAM8 (2K)	@ **£24.50**	PERICON A	@ **£27.90**
RAM16	@ **£26.75**	PERICON B	@ **£33.75**
RAM64	@ **£76.25**	PERICON C	@ **£41.75**

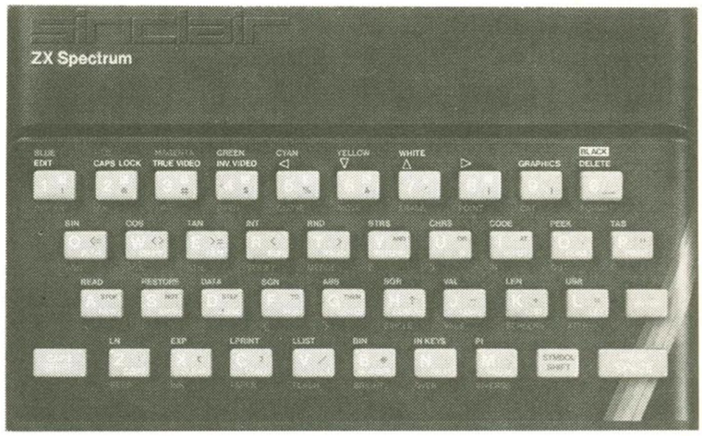
Options: USERFONT @ £8.00,
Add. 2K for RAM8 @ **£6.50,** Add. 2K for DROM @ **£7.50**

All prices include VAT, postage and packing in the U.K. (Overseas allow at least 15% for surface mail).

Post today to: BASICARE MICROSYSTEMS LTD., 5 Dryden Court, London SE11 4NH or Phone: 01-735 6408.

BASICARE MICRO SYSTEM

Der kleine Schwarze mit dem bunten Band



Der ZX-Spectrum ist da! Lange hat es gedauert. Lieferzeiten bis zu 6 Monaten waren an der Tagesordnung, doch jetzt ist es endlich soweit, daß es sich lohnt, den Neuen aus dem Hause Sinclair näher zu betrachten.

Um es gleich vorwegzunehmen; alles was der ZX-Spectrum kann im Detail zu schildern, würde den Rahmen dieses Magazins bei weitem sprengen. Wir können hier lediglich auf die technischen Daten, seinen Befehlssatz und auf einige Besonderheiten hinweisen. Deshalb haben wir auch erst gar nicht auf der Titelseite großspurig 'Alles über den ZX-Spectrum' getönt. Man kann leider nur sehr wenig über ein Gerät aussagen, das man erst einige Tage im Hause hat und in dessen Feinheiten man sich erst in Monaten eingearbeitet haben wird. Dann freilich wird der Homecomputer-Leser sich selbst ein genaues Bild machen können, denn in der Zwischenzeit werden wir die ersten grösseren Programme zum Spectrum veröffentlicht haben und so etwas mehr über die Möglichkeiten rund um den Spectrum vermitteln können.

Zur Sache: Der ZX-Spectrum wird in einer Größe von 23,3 x 14,4 x 3,0 cm geliefert und verfügt über einen Z80A-Microprozessor, der mit einer Taktfrequenz von 3.5 MHz seinen Dienst verrichtet. Sein BASIC-Interpreter und das Operating-System sind in 16 K-Byte ROM untergebracht. Den dem Benutzer für Programme und Daten zur Verfügung stehende Bereich kann man mit 16 KByte oder 48 KByte RAM bestückt erhalten. Eine 32-KByte-Aufrüstplatine soll später zu haben sein, falls man sich zuerst für die kleinere Ausführung entschieden hat. Zum ersten Mal bietet Sinclair seinen Kunden eine Schreibmaschinentastatur, obwohl man das richtige Schreibmaschinengefühl noch nicht erlebt, wenn man die 40 beweglichen 'Gummipuffertasten' bearbeitet. Aber teure Mechanik müßte eben auch teuer bezahlt werden und deshalb wollen wir das auch nicht überbewerten. Die Freude an der Arbeit mit diesem Gerät wurde uns jedenfalls dadurch nicht getrübt.

Gewöhnen muß man sich allerdings an die Art wie der Spectrum zu programmieren ist. Die meisten Tasten haben fünf, einige sechs verschiedene Funktionen, und es dauert schon eine Zeit, bis man das alles beherrscht, vor allem für den, der sonst, sei es in der Uni oder im Betrieb, größere Rechner bedient. Groß/Kleinschrift ist ebenso vorhanden, wie der volle ASCII-Zeichensatz (im Gegensatz zum ZX-81 ist dieser international codiert!), eine Lowresolutiongrafik, 22 Farbkommados, 21 anwenderprogrammierbare Grafikzeichen und Wiederholfunktion aller Tasten (Auto-Repeat).

Mit dem ZX-Spectrum vefügt man über Möglichkeiten hochauflösende Graphik zu programmieren, die bei einer 256x192-Punktauflösung für Spiele und nicht professionelle Anwendungen vollkommen ausreicht. Sie benötigt ca. 7 K vom verfügbaren RAM und es ist von daher vorteilhaft, sich gleich die 48-K-Version zuzulegen, wenn man auf die Graphik besonderen Wert legt. Für jeden von den 768 Zeichenpunkten kann eine von 8 Farben als Vordergrund, als Hintergrund, in einfacher oder besonderer Helligkeit und Blinken oder Normaldisplay definiert werden. Ebenso ist die Farbe des Bildschirmrandes in den gleichen Farben programmierbar.

Obwohl sich einzelne Punkte in Hochauflösung setzen lassen, kann die Farbe mehrerer Einzelpunkte aber innerhalb eines Zeichenblocks nur gleich sein, was für die vorkommenden Anwendungen im Heimbereich aber genügen dürfte. Echte Highresolution-Farbgraphik hat man demnach nicht.

Punkte, Linien, Kreise und Winkel zu zeichnen ist aufgrund vorhandener BASIC-Kommandos einfach, Text und Graphik sind untereinander mischbar, was wiederum Vorteile sind, die nur wenige andere größerere Micros mitbringen.

Anschluß ist an jedes PAL-Color- oder S/W-Fernsehgerät möglich (UHF Kanal 36). Bei Verwendung eines S/W-Fernsehers erzeugen die Farben eine Grauskala!

Die BASIC-Worte für die Farbprogrammierung sind:
INK (Vordergrund)
PAPER (Hintergrundfarbe)
BRIGHT (Helligkeit) und
FLASH (Blinken).

OVER schreibt über den Bildschirm ohne vorhandene Graphik oder Texte zu löschen.

INVERSE sorgt für eine Umkehrung der Farbinformationen (Vordergrundfarbe wird Hintergrund und umgekehrt). Sämtliche Funktionen sind über das Tastenfeld und in Programmen zugänglich.

Die acht möglichen Farben sind Schwarz, Blau, Rot, Magenta (Lila), Grün, Cyan (Hellblau), Gelb und Weiß.

Der ZX-Spectrum hat einen eingebauten Lautsprecher; aber auch einen Anschluß

Die ZX-Seiten

für externen Lautsprecher & HiFi-Anlage. Über ein BEEP-Kommando sind 130 Halbtöne - mehr als 10 Oktaven! - Musik zu erzeugen.

Das Bildschirmfenster ist unterteilt in 2 Sektionen. Im oberen Teil (normalerweise 22 Linien) erscheint das Programmlisting oder die Ausführung des Programms. Der untere Teil (normal 2 Linien) zeigt die gerade eingegebene oder editierte Programmzeile, zeigt Statusmeldungen, usw. an. Auch das Editieren geschieht, unter Verwendung aller üblicher Cursorfunktionen, incl. INSERT und DELETE, im unteren Teil des Bildschirmes. Dieser vergrößert sich wenn nötig, um bis zu 22 Zeilen (=704 Zeichen) in eine Programmzeile aufzunehmen!

+, -, *, /, ↑, SIN, COS, TAN, ASN, ACS, ATN, LN, EXP, SGN, ABS, INT, SQR, RND und PI sind vorhandene mathematische Funktionen. Der Zahlenbereich 3E-39 bis 7E+38 (Floating Point) ist auf 9 1/2 Stellen genau. Die Direkteingabe von Binärzahlen ist möglich. Strings und Zahlenwerte können mit =, >, <, >=, <= und <> verglichen werden. Weitere logische Funktionen sind AND, OR und NOT. 52 anwenderprogrammierbare Funktionen können mit DEF FN (26 numerische und 26 Stringargumente) zur Erzeugung von numerischen und Stringergebnissen definiert werden.

Mit READ, DATA und RESTORE können auch mit dem ZX-Spectrum endlich auch übliche BASIC-Programme erarbeitet werden. ZX-81-Besitzer werden es begrüßen.

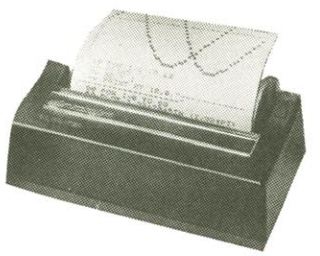
Überhaupt ist das Sinclair-Basic äußerst vielseitig - wenn auch etwas langsam (was andererseits durch einen preisgünstigen lieferbaren Compiler zu beheben ist), wie der Anwender des ZX-Spectrum feststellen wird. Es ist zum Beispiel auch in puncto Stringbearbeitung nichts ausgelassen, was die "Großen" haben und einiges mehr implementiert.

a$(x TO y) ist solch ein Zusatz mit dem Teile aus Strings herausgeholt werden können. Kaum nötig zu erwähnen, daß sowohl Stringarrays als auch numerische mehrdimensionale Felder zu programmieren sind.

Während der Programmausführung wird laufend ein 'Expression evaluator', ein Ausdruck-Auswerter, abgerufen, sobald ein Ausdruck, eine Variable oder Konstante erreicht wird. Dies ermöglicht das Programmieren von GOTO und GOSUB Label und ermöglicht außerdem den Gebrauch des ZX-Spectrum als normalen Rechner.

Auch das Kassetteninterface ist fortschrittlich konzipiert, wie der Rest des Spectrum. Vor jeder Aufnahme wird zunächst ein Führungston auf Band gebracht, um die bei manchen Kassettenrecordern übliche Level-Fluktuation auszugleichen und ein Schmitt-Trigger beseitigt Rauschen beim Playback. Alle Informationen werden vor dem Abspeichern mit einem Datenkopf versehen, der Informationen über Filetyp, Filenamen, Länge und Adressen enthält.

Programme, Bildschirme, Memoryblocks, String- und FP-Arrays können gezielt gespeichert und abgerufen werden. Danach kann man sie auf Fehler untersuchen, um sicheres fehlerfreies Speichern zu gewährleisten. Später ist es möglich, Programme und Arrays vom Band zu 'mergen' und sie so mit im Rechner vorhandenen Daten und Programmen zu kombinieren. Eine weitere Neuerung im Microcomputerbereich ist das Retten von Programmen zusammen mit einer Zeilennummer, mit der das Programm nach dem Laden gestartet wird. Der Datentransfer hat eine Geschwindigkeit von 1500 Baud und geschieht somit schneller als bei den meisten Konkurrenten.



Der vom ZX-81 bekannte ZX-Drucker ist voll kompatibel und druckt auch Kleinbuchstaben und die Highresolution-Grafik aus.

Auch für schnelleren und gezielteren Datenaustausch soll gesorgt werden. Wir haben ihn zwar noch nicht, doch in England sollen schon die ZX-Microdrives (von Sinclair entwickelte Micro-Diskettenstationen) lieferbar sein.

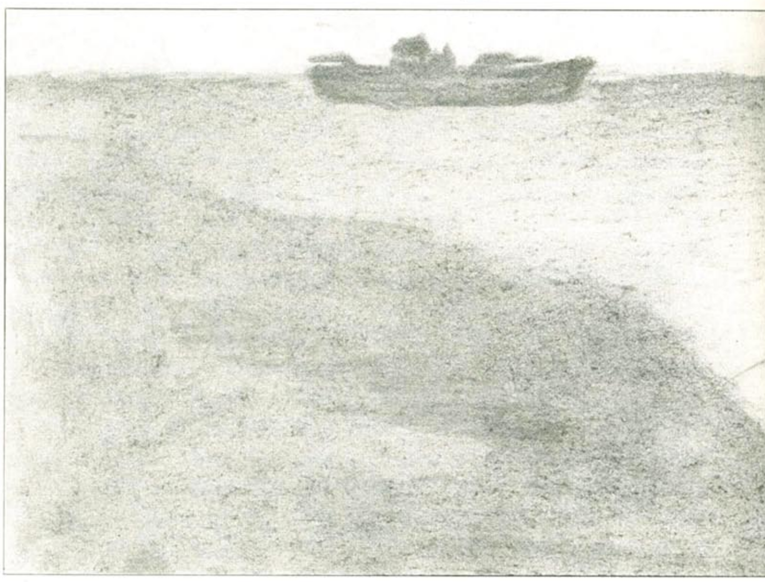
Auf verschiedenen Ausstellungen dort wurden sie schon vorgestellt und sie sollen tatsächlich nur ca. 250,-DM pro Stück - acht können angeschlossen werden - kosten.

Der schon bekannte ZX-Drucker ist ebenfalls kompatibel und druckt auch die Highresolutionscreen und die Kleinbuchstaben aus.

Bisheriges Fazit:

Wer immer es mit einem preiswerten Computersystem versuchen möchte: Schüler, Studenten, Vereine, Sportclubs, Familien und Kleinstbetriebe, sollten eine Anschaffung des ZX-Spectrum durchaus in Erwägung ziehen. Es ist ein Riesen-Computer zu einem Zwergenpreis.

ZX-U-Boot-Jagd



Dieses flackerfreie Spielprogramm wurde in der Maschinensprache für den 1K-ZX-81 geschrieben. Es läuft wegen des Hintergrundausdruckes nur auf Maschinen bis 3.5K (eventuell 16K-RAM-Pack abziehen). Ziel des Spieles ist es, ein Unterseeboot in schnellstmöglicher Zeit mit einem Torpedo zu treffen. Mit den Tasten 5 & 8 kann das Schiff des Spielers bewegt werden (es ist doppelt so schnell wie das U-Boot) und mit der Taste 0 wird der Torpedo ausgelöst.

Innerhalb von 25 Sekunden zählt ein Zähler von 999 an abwärts. Das Spiel ist zu Ende, wenn dieser Zähler auf Null steht oder wenn das U-Boot getroffen wurde.

```
U-BOOT-JAGD  *  ZX-81

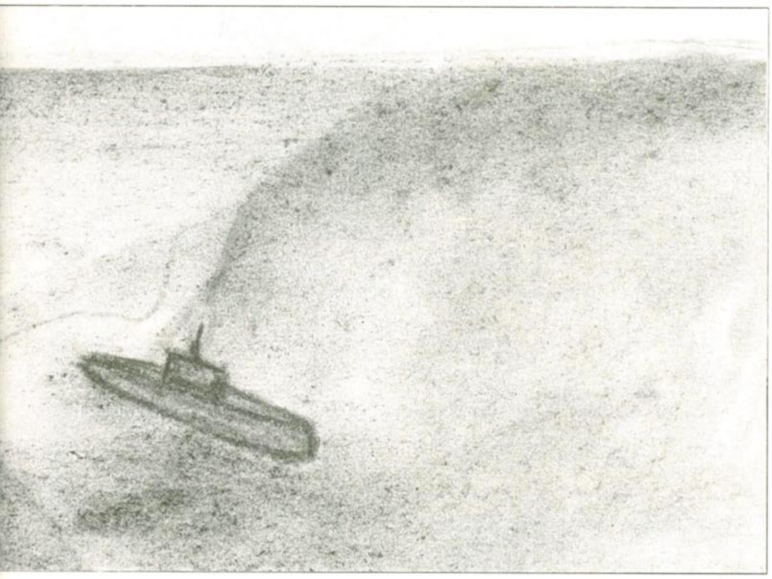
 16514     6    6   62  118  215   16  253  175
 16522   215   62   37  215  215  215   62  118
 16530   215  215    6   64   62   14  215   16
 16538   253   33   64    1   62  136  215   43
 16546   124  181   32  248    6   32   62  128
 16554   215   16  253   42   12   64   17   74
 16562     0   25   34   60   64   33    0    1
 16570    34   68   64   42   50   64   35   34
 16578    50   64  125  230    7  206    5  205
 16586    25   21   62   33  205   25   21  239
 16594     4   52  205  167   14   42   12   64
 16602     9   17    9    0   25   34   64   64
 16610     1    0    0   42   68   64   35   34
 16618    68   64   42   68   64   43  124  181
 16626    32  251   42   12   64   17   10    0
 16634    25  126  167   32    8    6    3   35
 16642    54   28   16  251  201   61  254   27
 16650    32    5   54   37   43   24  234  119
 16658     4  203   64   32  102   42   64   64
 16666   126  254  187  202  182   65   54  136
 16674    35   54  136   35   54  136   43   43
 16682    43  126  254  118   40  141   34   64
 16690    64   54  138   35   54  177   35   54
 16698   138  175  185   32   19   58   38   64
 16706   254  253   32   56   14    1   42   60
 16714    64   17   34    0   25   34   66   64
 16722    42   66   64   54  136   17   33    0
 16730    25  126  254  128   32    4   14    0
 16738    24   26  126  254  177   40   77   42
 16746    64   64   43  126  254  118   40  238
 16754    42   66   64   17   33    0   25   54
 16762   187   34   66   64   42   60   64   54
 16770    14   35   54   14   35   54   14   58
 16778    38   64  254  223   32   11   43   43
 16786    43  126  254  118   32   16   35   24
 16794    13  254  247   32    7   35  126  254
 16802   118   32    1   43   43   43   34   60
 16810    64   54  134   35   54  187   35   54
 16818     6  195  229   64   42   64   64   54
 16826   167   35   54  180   35   54  180   35
 16834    →                         54  178  201

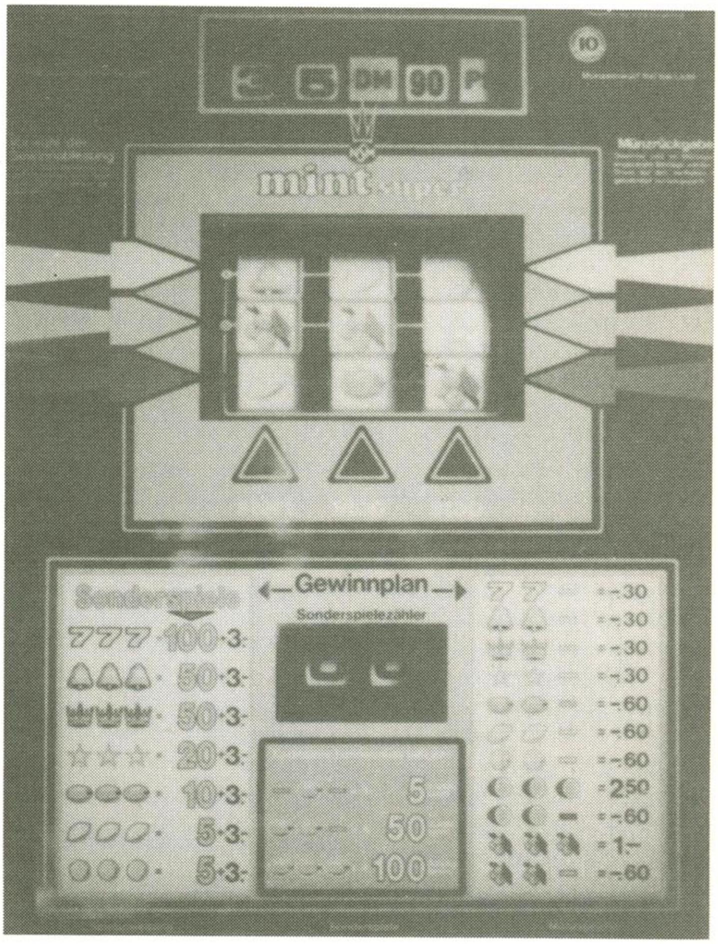
U-BOOT-JAGD  *  ZX-81

1  REM  000   (323NULLEN)   000
10  LET X = 16514
20  INPUT A
30  POKE X,A
40 X = X + 1
50  GOTO 20

]

10 IF NOT USR 16514 THEN PRINT " VERFEHLT ! ! !"
```





ROTAMINT-

der Geldspielautomat

Das Programm ROTAMINT simuliert einen Geldspielautomaten der Super-Sieben-Serie. Dem Original wird im Aufbau, Spielablauf und Gewinnplan weitgehend entsprochen.

Nach Eintasten des Maschinenprogrammladers wird das Maschinenprogramm in Dezimal in das REM-Statement in Zeile 1 gePOKEd. Dann werden alle Zeilen ab 10 gelöscht und stattdessen die Zeile:

10 IF NOT USR 16514 THEN PRINT"VERFEHLT"

eingetastet. Nun sicherheitshalber zuerst das gesamte Programm auf Kassette retten, damit es im Falle einer falschen Eingabe, die selbstzerstörerisch wirken könnte, neu geladen und mittels PEEKen überprüft werden kann.

Das Programm wird mit RUN im Direktmodus gestartet.

Auf den Walzen von ROTAMINT sind insgesamt 9 verschiedene Symbole verteilt. Als deren Stellvertreter erscheinen in den Fenstern des Bildschirms große Buchstaben bzw. eine Zahl. Vorgängerfeld und Nachfolgerfeld werden jeweils oberhalb und unterhalb der Fenster angezeigt.

Nr.	Symbol	Stellvertreter
1	SIEBEN	7
2	GLOCKE	G
3	KRONE	D
4	STERN	S
5	PFLAUME	P
6	ZITRONE	Z
7	ORANGE	O
8	MELONE	M
9	KIRSCHE	K

Die ZX-Seiten

Die Walzen halten in der Reihenfolge 1,3,2 an. Walze 1 kann einmal mit einer Starttaste erneut in Drehung versetzt werden. Die Walzen 2 und 3 können mit einer Stoptaste angehalten werden. Sie sind an einen Mischer gekoppelt, der die Walzen mit einer zufälligen Verzögerung reagieren läßt. Die Variable MISCHER legt dafür einen Verzögerungsspielraum fest. So bleibt die Walze bei MISCHER=3 spätestens drei Felder nach Drücken der Stoptaste stehen, während MISCHER=0 die Walze sofort reagieren läßt.

Als Information über die momentane Stellung der Walze dient das Orientierungsfeld. Es zeigt während der Rotation durch kurzes Aufblinken zweier Punkte an, daß es sich gerade im Fenster befindet. Das Orientierungsfeld bleibt über einen längeren Zeitraum unverändert. Danach wird in unregelmäßigen Abständen ein anderes Feld zum Orientierungsfeld.

ROTAMINT hat zu Beginn einen Inhalt von 100,- DM. Nach Aufnahme des Spielbetriebs durch Tastendruck wird zunächst der Münzeinwurf aktiviert. Im weiteren Verlauf geschieht dies immer dann, wenn sich nicht mehr genügend Geld für ein Spiel (30 Pfg.) im Speicher befindet. Die Tasten 1-9 werden als DM-Beträge interpretiert, die Taste 0 als Groschen.

Eine Auszahlmöglichkeit von Gewinnen besteht nach jedem fünften Spiel.

Bei jeweils drei gleichen Symbolen gibt es für SIEBEN bis ORANGE 3,-DM und 100, 50, 50, 20, 10, 5 bzw. 5 Sonderspiele. M-M-M und K-K-K erzielen 2.50 DM bzw 1.- DM und keine Sonderspiele. Nur zwei gleiche Symbole im mittleren Feld und linken Fenster ergeben für PFLAUME bis KIRSCHE 30 Pfennige und sonst 60 Pfennige.

In einer Sonderspielserie mit mehr als 20 Spielen sind weitere Serien ausgeschlossen. Für das letzte Spiel aller anderen Serien gilt außerdem folgende Regelung: drei PFLAUMEN ergeben 100 zusätzliche Sonderspiele, zwei links und in der Mitte 50 und nur eine in der Mitte 5 Sonderspiele.

In einer Serie werden alle Gewinne auf 3,- DM erhöht.

Änderungen des Gewinnbetrags können in der Einheit Groschen vorgenommen werden. So legt:

LET G$(1,2,3) = CHR$15

einen Gewinn von 1,50 DM für die Kombination 7-G-K fest.

Let E(9)=10 führt zu 10 Sonderspielen für drei Kirschen.

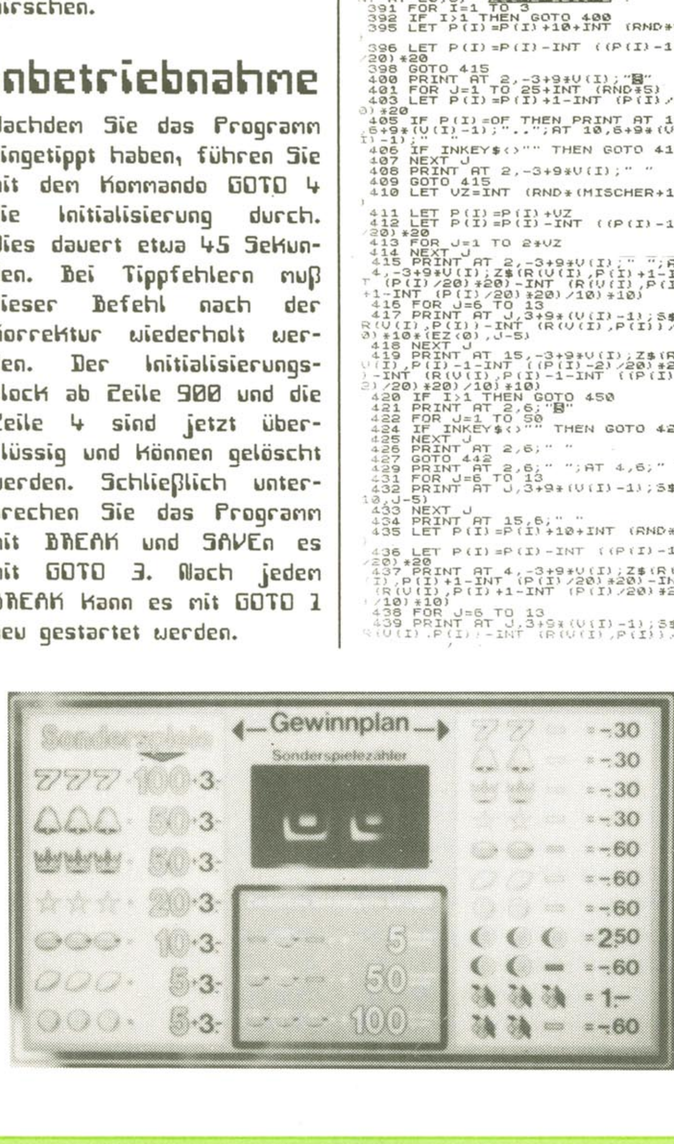
Inbetriebnahme

Nachdem Sie das Programm eingetippt haben, führen Sie mit dem Kommando GOTO 4 die Initialisierung durch. Dies dauert etwa 45 Sekunden. Bei Tippfehlern muß dieser Befehl nach der Korrektur wiederholt werden. Der Initialisierungsblock ab Zeile 900 und die Zeile 4 sind jetzt überflüssig und können gelöscht werden. Schließlich unterbrechen Sie das Programm mit BREAK und SAVEn es mit GOTO 3. Nach jedem BREAK kann es mit GOTO 1 neu gestartet werden.



```
GEWINN   3,00  DM
SUMME   35,80  DM
SONDERSPIELE   13
```

```
   2 GOTO 40
   3 SAVE "MINT"
   4 GOSUB 900
   5 REM PARAMETER*********
   6 LET INHALT=10000
   7 LET MISCHER=3
   8 LET OF=20
   9 LET WH=50
  10 REM ******************
  11 LET IH=INHALT
  12 LET GB=0
  13 LET SZ=0
  14 LET EZ=0
  20 RAND
  40 PRINT AT 1,2;"GELDSPIELAUTOMAT G-E-S-P-A"
  50 PRINT AT 4,6;"INHALT";TAB 14;INT (IH/100);",";(IH-INT (IH/100)*100)/10;"0  DM"
  55 PRINT AT 17,22;"SPIEL"
  62 FOR J=1 TO 10
  64 PRINT AT 17,30;" "
  68 IF INKEY$<>"" THEN GOTO 300
  70 PRINT AT 17,30;"▒"
  72 NEXT J
  75 GOTO 60
 300 CLS
 301 FOR I=2 TO 29
 305 PRINT AT 5,I;"▒";AT 14,I;"▒"
 310 NEXT I
 320 FOR I=6 TO 13
 325 PRINT AT I,2;"▒";TAB 11;"▒";TAB 20;"▒";TAB 29;"▒"
 330 NEXT I
 340 FOR I=7 TO 24
 345 PRINT AT 17,I;"▄";AT 21,I;"▀"
 350 NEXT I
 360 FOR I=18 TO 20
 365 PRINT AT I,7;"▌";TAB 24;"▐"
 370 NEXT I
 380 PRINT AT 18,8;"GEWINN";TAB 22;"DM";AT 19,8;"SUMME"
 382 PRINT AT 19,16-(GB>=1000);INT (GB/100);",";(GB-INT (GB/100)*100)/10;"0  DM"
 383 IF EZ>0 THEN PRINT AT 20,8;"SONDERSPIELE";TAB 23-(EZ>9);EZ
 384 IF GB<30 THEN GOTO 472
 390 IF EZ=1 AND AS<=20 THEN PRINT AT 20,8;" ZUSAETZLICHE ? "
 391 FOR I=1 TO 3
 392 IF I>1 THEN GOTO 400
 395 LET P(I)=P(I)+10+INT (RND*8)
 396 LET P(I)=P(I)-INT ((P(I)-1)/20)*20
 398 GOTO 415
 400 PRINT AT 2,-3+9*V(I);"▒"
 401 FOR J=1 TO 25+INT (RND*5)
 403 LET P(I)=P(I)+1-INT (P(I)/20)*20
 405 IF P(I)=OF THEN PRINT AT 10,6+9*(V(I)-1);"..";AT 10,6+9*(V(I)-1);"  "
 406 IF INKEY$<>"" THEN GOTO 410
 407 NEXT J
 408 PRINT AT 2,-3+9*V(I);" "
 409 GOTO 415
 410 LET VZ=INT (RND*(MISCHER+1))
 411 LET P(I)=P(I)+VZ
 412 LET P(I)=P(I)-INT ((P(I)-1)/20)*20
 413 FOR J=1 TO 2*VZ
 414 NEXT J
 415 PRINT AT 2,-3+9*V(I);" ";AT 4,-3+9*V(I);Z$(R(V(I),P(I)+1-INT (P(I)/20)*20)-INT (R(V(I),P(I)+1-INT (P(I)/20)*20)/10)*10)
 416 FOR J=6 TO 13
 417 PRINT AT J,3+9*(V(I)-1);S$(R(V(I),P(I))-INT (R(V(I),P(I))/10)*10*(EZ<0),J-5)
 418 NEXT J
 419 PRINT AT 15,-3+9*V(I);Z$(R(V(I),P(I)-1-INT ((P(I)-2)/20)*20)-INT (R(V(I),P(I)-1-INT ((P(I)-2)/20)*20)/10)*10)
 420 IF I>1 THEN GOTO 450
 421 PRINT AT 2,6;"▒"
 422 FOR J=1 TO 50
 424 IF INKEY$<>"" THEN GOTO 423
 425 NEXT J
 426 PRINT AT 2,6;" "
 427 GOTO 442
 429 PRINT AT 2,6;" ";AT 4,6;" "
 431 FOR J=6 TO 13
 432 PRINT AT J,3+9*(V(I)-1);S$(10,J-5)
 433 NEXT J
 434 PRINT AT 15,6;" "
 435 LET P(I)=P(I)+10+INT (RND*8)
 436 LET P(I)=P(I)-INT ((P(I)-1)/20)*20
 437 PRINT AT 4,-3+9*V(I);Z$(R(V(I),P(I)+1-INT (P(I)/20)*20)-INT (R(V(I),P(I)+1-INT (P(I)/20)*20)/10)*10)
 438 FOR J=6 TO 13
 439 PRINT AT J,3+9*(V(I)-1);S$(R(V(I),P(I))-INT (R(V(I),P(I))/10)*10*(EZ<0),J-5)
 440 NEXT J
 441 PRINT AT 15,-3+9*V(I);Z$(R(V(I),P(I)-1-INT ((P(I)-2)/20)*20)-INT (R(V(I),P(I)-1-INT ((P(I)-2)/20)*20)/10)*10)
 442 IF EZ>=0 THEN LET EZ=EZ-1
 443 LET GB=GB-30
 445 PRINT AT 19,15-(GB>=1000);" ";INT (GB/100);",";(GB-INT (GB/100)*100)/10;"0"
 447 IF EZ=0 AND AS>20 THEN PRINT AT 20,8;"                "
 449 IF EZ>0 THEN PRINT AT 20,22-(EZ>9);" ";EZ
 450 LET S(V(I))=R(V(I),P(I))
 451 NEXT I
 452 LET B=CODE G$(S(1),S(2)-INT (S(2)/10)*10,S(3))*10
 453 IF EZ>=0 AND (B>0 OR S(2)>10) THEN LET B=300
 454 IF S(1)<>S(2)-(S(2)>9)*10 OR S(1)<>S(3) OR AS>20 AND EZ>=0 OR S(1)=5 AND EZ=0 THEN GOTO 457
 455 LET AS=E(S(1))
 456 LET EZ=EZ+AS+(EZ=-1)
 457 IF EZ<>0 OR AS>20 THEN GOTO 461
 458 LET AS=5*(S(2)=5)+45*(S(1)=5 AND S(2)=5)+50*(S(1)=5 AND S(2)=5 AND S(3)=5)
 459 LET EZ=EZ+AS
 460 IF EZ>100 THEN LET EZ=100
 461 IF EZ>0 THEN PRINT AT 20,8;AT 20,23-(EZ>9);EZ
 462 IF B THEN PRINT AT 18,16;INT (B/100);",";(B-INT (B/100)*100)/10;"0"
 463 LET GB=GB+B
 464 IF B THEN PRINT AT 19,14;"    ";AT 19,16-(GB>=1000);INT (GB/100);",";(GB-INT (GB/100)*100)/10;"0"
 465 PRINT AT 4,0;D$
 466 FOR J=6 TO 13
 467 PRINT AT J,3;"        ▒        ▒"
 468 NEXT J
 469 PRINT AT 15,0;D$;AT 18,14;"        ";AT 20,9;("               " AND EZ=0)
 471 IF GB>=30 THEN GOTO 495
 472 PRINT AT 0,17;"MUENZEINWURF"
 473 FOR I=1 TO 20
 474 PRINT AT 0,31;" "
 475 PRINT AT 0,31;"▒"
 476 LET C$=INKEY$
 477 IF C$<>"" THEN GOTO 480
 478 NEXT I
 479 GOTO 490
 480 IF C$=CHR$ 118 THEN GOTO 490
 481 IF C$<"0" OR C$>"9" THEN GOTO 473
 482 LET B=(CODE C$-28)*100
 483 IF NOT B THEN LET B=10
 484 IF GB+B>=8500 THEN GOTO 490
 485 LET GB=GB+B
 486 PRINT AT 19,16-(GB>=1000);INT (GB/100);",";(GB-INT (GB/100)*100)/10
 487 LET IH=IH+B
 488 GOTO 473
 490 PRINT AT 0,0;D$
 491 IF GB>=30 THEN GOTO 390
 492 CLS
 493 GOTO 50
 495 LET SZ=SZ+1
 496 IF SZ/WH<>INT (SZ/WH) THEN GOTO 500
 497 LET OF=INT (20*RND)+1
 498 LET OF=OF-INT ((OF-1)/20)*20
 499 LET WH=20+INT (RND*21)
 500 IF SZ/5<>INT (SZ/5) THEN GOTO 390
 502 IF GB>=IH THEN PRINT AT 2,0;"SUMME UEBERSTEIGT INHALT"
 503 PRINT AT 0,0;"AUSZAHLUNG"
 504 FOR I=1 TO 20
 505 PRINT AT 0,11;" "
 506 PRINT AT 0,11;"▒"
 507 LET C$=INKEY$
 508 IF C$<>"" THEN GOTO 511
 509 NEXT I
 510 IF GB<8500 THEN GOTO 546
 511 IF C$=CHR$ 118 AND GB<8500 THEN GOTO 546
 512 IF GB<2000 OR IH<1000 THEN GOTO 520
 513 LET GB=GB-1000
 514 LET IH=IH-1000
 515 GOTO 544
 520 IF GB<200 OR IH<100 THEN GOTO 530
 522 LET GB=GB-100
 523 LET IH=IH-100
 524 GOTO 544
 530 IF GB THEN GOTO 535
 531 FOR I=1 TO 50
 532 NEXT I
 533 GOTO 546
 535 IF IH THEN GOTO 542
 536 FOR I=1 TO 40
 537 PRINT AT 2,0;"AUTOMAT LEER           ";AT 2,0;"AUTOMAT LEER"
 538 NEXT I
 539 GOTO 546
 542 LET GB=GB-10
 543 LET IH=IH-10
 544 PRINT AT 19,15-(GB>=1000);" ";INT (GB/100);",";(GB-INT (GB/100)*100)/10
 545 GOTO 504
 546 PRINT AT 0,0;D$+D$+D$
 550 GOTO 471
 900 FAST
 902 LET D$="                                "
 903 DIM R(3,20)
 905 LET C$="198692897839859489791968295968397948597919869287583986947978"
 910 FOR I=0 TO 2
 912 FOR J=1 TO 20
 914 LET R(I+1,J)=VAL C$(I*20+J)+10*(I=1 AND INT (J/2)=J/2)
 915 NEXT J
 918 NEXT I
 932 LET Z$="7GOSPZOMK"
 940 DIM S$(19,8,8)
 950 FOR I=1 TO 9
 952 FOR J=1 TO 8
 954 LET D=PEEK (7679+J+CODE Z$(I)*8)
 956 LET C=256
 958 FOR K=1 TO 8
 960 LET C=C/2
 962 LET S$(I,J,K)=CHR$ (INT (D/C)*128)
 963 LET S$(I+10,J,K)=CHR$ (INT (D/C)*136)
 964 LET D=D-INT (D/C)*C
 966 NEXT K
 968 NEXT J
 970 NEXT I
 973 DIM G$(9,9,9)
 975 FOR I=1 TO 9
 977 FOR J=1 TO 9
 978 LET G$(I,I,J)=CHR$ (3+3*(I>1 AND I<>J))
 979 NEXT J
 980 LET G$(I,I,I)=CHR$ 30
 981 NEXT I
 982 LET G$(8,8,8)=CHR$ 25
 983 LET G$(9,9,9)=CHR$ 10
 985 DIM E(9)
 986 LET C$="KAA4211"
 987 FOR I=1 TO 7
 988 LET E(I)=(CODE C$(I)-28)*5
 989 NEXT I
 995 DIM V(3)
 996 LET V(1)=1
 997 LET V(2)=3
 998 LET V(3)=2
 999 DIM P(3)
1000 DIM S(3)
1002 LET AS=0
1005 SLOW
1010 RETURN
```



ZX81 GAMES

from

J.K. GREYE SOFTWARE LTD

THE NEW GENERATION SOFTWARE HOUSE

"Without question the finest machine code games available today"......... J.N. ROWLAND Product Manager for W.H. SMITH.

GAMESTAPE 1 for 1K ______ only £3.95
10 Games incl. ASTEROIDS, UFO, CODE, BOMBER, GUILLOTINE, KALEIDESCOPE, etc.
PROBABLY THE BEST VALUE 1k TAPE AVAILABLE.



We've done in 1k, games which some of our competitors require 16k to do!



GAMESTAPE 2 for 16K ______ only £3.95
***STARFIGHTER** Superb machine code Space Battle. Set against a background of twinkling stars, with stunning explosions — if you can hit the enemy!
PYRAMID Can you move the Pyramid? Make a mistake and it will collapse! A Thinkers game.
ARTIST The ultimate Graphic Designers aid. 8 Directions, 10 Memories, SAVE, COPY, RUBOUT, CLS, etc.

GAMESTAPE 3 for 16K ______ only £4.95
***CATACOMBS** A Multi-Level Graphics Adventure. Each level can contain up to 9 Rooms, 8 Passages, 7 Monsters, Food, Gold, Traps, Phantoms, an Exit (to the next level), and there's an infinite number of levels.
NOTE. . . . This is NOT one of the necessarily limited text Adventures as sold elsewhere.
"An excellent addictive game which will keep you amused for hours.". . . . COMPUTER & VIDEO GAMES.

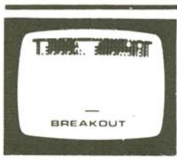




GAMESTAPE 4 for 16K ______ only £4.95
***3D MONSTER MAZE** The Game to Top All Others. Unbelievable Graphics! Can you find your way through the Maze? The EXIT is there somewhere, but then so is a T.REX, and its after YOU! All in 3D (the T.REX will actually run towards you in full perspective!), you've never seen anything like this before!
"3D MONSTER MAZE is the best game I have seen for the ZX81". . . COMPUTER & VIDEO GAMES
"If I had to choose just one programme to impress an audience with the capabilities of the ZX81, then J.K. Greye's 3D MONSTER MAZE would be the one without doubt" . . . ZX COMPUTING. *"Brilliant, brilliant, brilliant!".... POPULAR COMPUTING WEEKLY*

GAMESTAPE 5 for 16K ______ only £4.95
***3D DEFENDER** The Ultimate Space Game. Super fast Machine Code 3D version of the Arcade favourite. You have to save your home planet from the marauding Alien Spacecraft. This is all in 3D, your viewscreen shows you the view out of your fighters cockpit window. The backdrop moves when you turn, or fly up or down (8 flight directions), just as if you were really flying it! But then YOU ARE! The Enemy Saucers will actually zoom towards you in 3D, and shoot you if you let them! Your display includes Score, Shield Strength, Altitude, Proximity, Forward Radar and your viewscreen, which shows your rotating home planet, backdrop of Stars, Meteors, Explosions, Plasma Blasts, your Photon Beams, up to 4 Enemy Saucers and of course its all in full 3D!
"Another 3D winner"....................................... SINCLAIR USER





GAMESTAPE 6 for 1K ______ only £1.95
***BREAKOUT** Super Fast Full Screen Display Game. Your all time favourite with an added twist. See how much Money you can win and watch the pounds convert to Dollars. All in Machine Code for Fast Action with 3 Speeds, 2 Bat Sizes and three angles of rebound! The best BREAKOUT around and at this price you can't go wrong!
"The best of its kind" .... WHICH MICRO & SOFTWARE REVIEW

GAMES MARKED * INCL. MACHINE CODE.
Prices include VAT and U.K. P. & P.
(Add appropriate Postage on Foreign Orders). Cheques/P.O.s to

J.K. GREYE SOFTWARE LTD

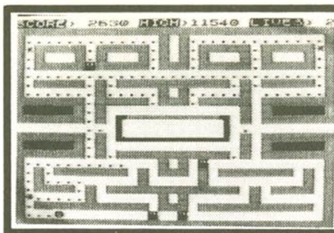
Dept HC 3, Brendan Close, Oldland Common, Bristol BS15 6QE

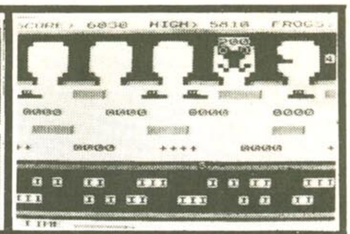
CREDIT CARD SALES: FOR INSTANT DESPATCH, **BY PHONE ONLY**
TEL: 01-930-9232 (9 am - 7 pm)

If you prefer to see before buying, our range of **GAMESTAPES** are stocked by the following stores.

BUFFER MICROSHOP	374A Streatham High Rd., London SW16:
GAMER	24 Gloucester Rd., Brighton;
GEORGES	89 Park St., Bristol, Avon;
MICROSTYLE	29 Belvedere, Lansdown Rd., Bath, Avon;
MICROWARE	131 Melton Rd., Leicester;
SCREEN SCENE	144 St. Georges Rd., Cheltenham, Glos;
W.H. SMITH	Over 200 Computer Branches;
ZEDXTRA	5 School Lane, Kinson, Bournemouth, Dorset:

TRADE & EXPORT ENQUIRIES WELCOME





ZUCKMAN
ZX81 (16K)
- *ALL MACHINE CODE (10K)
- *FOUR INDEPENDENT GHOSTS
- *HIGH-SCORE 'HALL OF FAME'
- *AUTHENTIC ARCADE ACTION
- *TITLE/DISPLAY MODE

ONLY £5.95 INC. P&P

FROGGER
ZX81 (16K)
- *MOVING CARS, LOGS, TURTLES
- *ALLIGATORS, DIVING TURTLES
- *FOUR 'SCREENS' OF ACTION
- *ALL ARCADE FEATURES
- *ENTIRELY MACHINE CODE

ONLY £5.95 INC. P&P

NEW ☆ **DRAGON 32 SOFTWARE** ☆ NEW

MISSILE COMMAND.......................... ONLY £5.95
FULL HIGH-RESOLUTION COLOUR GRAPHICS + SOUND

GRAPHICS DEMONSTRATOR........... ONLY £5.95
LEARN THE SECRETS OF HI-RES GRAPHICS PROGRAMMING

DJL SOFTWARE
DEPT HC, 9 TWEED CLOSE, SWINDON, WILTS SN2 3PU

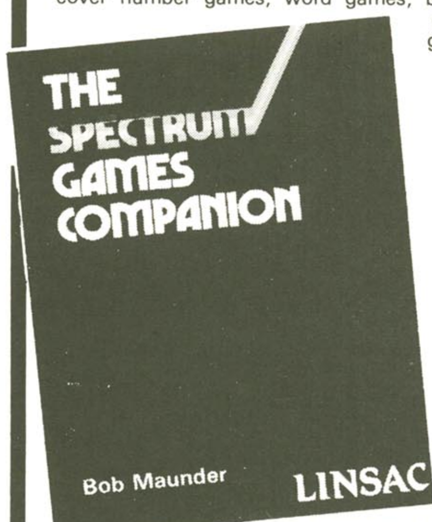
TRADE ENQUIRIES WELCOME

Linsac's ZX Companion series has received excellent press reviews:

"Far and away the best" — *Your Computer*

Thoughtfully written, detailed and illustrated with meaningful programs ... outstandingly useful" — *EZUG*

'The Spectrum Games Companion' is the latest addition to the series and is aimed at the games player and programmer alike. Twenty-one games designed specifically for the ZX Spectrum are included, with clear instructions on entry and play. Each program is explained fully with complete details on how it is designed and written. Introductory chapters show how to set up and use the Spectrum and how to create your own games. Later sections cover number games, word games, board games, simulation games, dice games, card games and grid games. If you want to enjoy your ZX Spectrum and learn its secrets at the same time then this is the book for you!



Bob Maunder is co-author of 'The ZX80 Companion' and author of 'The ZX81 Companion'. He is a Senior Lecturer in Computer Science at Teesside Polytechnic, holds an MSc degree in Computer Science, and is a Member of the British Computer Society.

The Spectrum Games Companion is available from good book shops, or send £5.95 to:

ONLY £5.95

LINSAC, (   ) 68 Barker Road, Middlesbrough, Cleveland TS5 5ES

ISBN 0 907211 02 X

Postage is free within the U.K. — add £1 for Europe or £2.50 outside Europe.

Apple II Disk-Handler

Das File-devellopment-program (FID) auf der System Master Diskette bietet die Möglichkeit, Files von einer Diskette auf eine andere zu kopieren. Weil die anderen Funktionen etwas umständlich zu handhaben sind, haben wir uns einige eigene Disk-Handling-Programme geschrieben. Nach und nach wurden es mehr und schließlich haben wir alles in einem Programm untergebracht, das sich lohnt, in Homecomputer vorgestellt zu werden.

Gleich am Anfang, damit es nicht überlesen wird, sei gesagt: Das Programm ist nicht für alle Disketten geeignet. Auf keinen Fall sollten Disketten mit Programmen, deren CATALOG normalerweise nicht auf dem Bildschirm dargestellt werden kann, vom Disk-Handler behandelt werden! Das Programm führt nämlich jedesmal, wenn eine Dikette aufgerufen wird, eine alphabetische Ordnung des Catalogs durch. Lediglich das Greeting-PGM wird nicht darin einbezogen. Wer das nicht wünscht, kann in Zeile 60 den Befehl CALL AD+6 löschen oder die 'Alphabetisierung' auf Tastendruck im Programm unterbringen.
Zum Programm als solches ist zu sagen, daß es ohne 'Beneath Apple-DOS' geschrieben wurde und deshalb einiges nach Studium dieser Lektüre anders und kürzer zu lösen gewesen wäre, was aber die Funktion nicht beeinträchtigt.
Besonders interessant ist es, die ganze Diskette sozusagen auf einen Blick zu betrachten. Das Programm bietet deshalb einen grafischen Ausdruck der belegten Sektoren. Dabei kann optisch erkannt werden, ob eine weitere Funktion, das Nutzbarmachen der Tracks 0-2 für Daten-Files, in Anspruch genommen wurde. Ca. 10% mehr Platz für das Abspeichern von Daten sind dadurch zu gewinnen; das DOS wird auf Daten-Disketten ohnehin nicht benötigt.

Die Funktionen im Einzelnen in der Reihenfolge ihres Auftretens im Titel.

<V>OL
Es wird die Volume-Nummer in der VTOC (siehe DOS-Handbuch) geändert, nicht jedoch die Volume-Nummer, die beim Initialisierungsvorgang unauslöschlich auf die Diskette gebracht wird und nur durch erneutes Initialisieren zu ändern ist. Allerdings ist auch nicht auszuschließen, daß der eine oder andere Leser etwas experimentiert hat und diese Behauptung widerlegen kann.

DO<S>
Der Diskettenbereich der Tracks 0-2 wird mit Daten gefüllt, die von einer anderen Diskette stammen. Dies ist interessant, wenn neuere Disketten schnellere DOS-Versionen oder sonstige Modifikationen enthalten und das DOS auf ältere übertragen werden soll, ohne den anderen Weg zu wählen, nämlich alle Files einzeln auf eine frisch initialisierte Disk zu übertragen.
Der Name des Greeting-Programms muß eventuell mit RENAME geändert werden.

<L>OAD
Wenn PG (Page) größer als Null ist, statt der Grafikdarstellung des Disketteninhalts ein Teil des Catalogs auf dem Bildschirm steht, dann kann durch L und den Kennbuchstaben ein gewünschtes Programm in den Speicher geladen werden und zwar wird vom Programm LOAD oder BLOAD selbst gewählt. In den meisten Fällen wird dabei das Programm DISK-HANDLER natürlich überschrieben werden.

<R>UN
Für RUN gilt dasselbe wie für LOAD nur, daß das gewählte Programm sofort gestartet wird.

< >NXTPG (Space drücken)
"Blättert" die Catalogseiten um.

LOC<K>
<U>NLOCK
<U> beziehungsweise <K> und ein Kennbuchstabe besorgen bequemes Sichern oder Entsichern der Files.

<K> oder <U> gefolgt von <K> hat zur Folge, daß sämtliche Files geLOCKed oder geUNLOCKed werden.

<F>ULL
Die Spuren, die das DOS enthalten werden freigestellt. Es sind nun 136 statt 124 KByte zur Aufnahme von Daten bereit.

RE<N>AME
Wenn das Programm genau eingetastet wurde, dann steht der Cursor nach Aufruf des zu ändernden Namens direkt unter diesem. Das wurde implementiert, weil in vielen Fällen nur ein Teil des Namens zu ändern ist und man deshalb mit <ESC> <I> und unter Verwendung der Taste Pfeil-nach-rechts bequem den neuen Namen eingeben kann.

<B>ACKUP
<C>OPY
Diese beiden Kommandos starten jeweils Programme von der System-Master-Disk, und zwar wird durch <B> das PGM COPYA aufgerufen und nach <C> befindet man sich im Programm FID.

<D>ELETE
Das Disk-File dessen Kennzeichen nach <D> aufgeru-

fen wurde, wird gelöscht. Sicherheitshalber wird aber vorher noch einmal nachgefragt, ob es sich tatsächlich um das zu löschende File handelt, dessen Name im Titel erscheint.

<G>RAPH
Der Disketteninhalt wird, wie schon erwähnt graphisch ausgedruckt. Für benutzte Sektoren wird ein "*" angezeigt und freie Sektoren werden durch einen Punkt herausgehoben. Dargestellt wird immer der Inhalt der Diskette, die zuletzt mit <W> aufgerufen wurde (dasselbe gilt übrigens auch für den Catalog).

NE<W>
Bringt den gesamten Catalog und die VTOC der Diskette in den Speicher des Apples und es können alle beschriebenen Funktionen verwendet werden.

<E>XIT
Beendet das Programm.

<0> <=> <->
Drei Befehle, die die Druckerausgabe steuern. Die Druckroutinen sind für einen MPI-Drucker ausgelegt, können aber ohne Schwierigkeiten für jeden anderen Drucker umgeschrieben werden.
Im Einzelnen bedeuten die Kommandos: <0> druckt eine Leerzeile, <=> druckt den gesamten Bilschirminhalt ab Zeile 4 aus. Und zwar bei Grafikdarstellung bis zum unteren Rand und bei Catalog-Display bis zur jeweils letzten auf dem Bildschirm angezeigten Zeile. <-> sorgt dafür, daß weitere Catalogseiten, direkt an einen vorhergehenden Ausdruck angeschlossen werden.
Programme hierzu:
Seite 39 - 41.

APPLE-STENO

Groß/Kleinschrift und so weiter...

Wie bringt man dem Apple Groß/Kleinschrift mit deutschem Zeichensatz bei? Man kauft ein Interface für 150,-DM und das Problem ist gelöst. Oder?

Bei mir war dies nicht der Fall. Nach Installation der Keyboard Platine (den Namen des Herstellers möchte ich hier nicht nennen, weil es durchaus sein kann, daß ausgerechnet meine Platine fehlerhaft ist und die Sache sonst fehlerfrei funktioniert), jedenfalls nachdem ich meine Platine eingebaut hatte, prellten erst einige dann mehrere, dann alle Tasten. Zuerst glaubte ich, ich hätte beim Einbau die Tastatur beschädigt. Erst als ich nach Anleitung den alten Zeichensatz zuschalten wollte, merkte ich, daß die Platine (bestückt mit 2 Widerständen und 1 Eprom am Tastenprellen Schuld hatte. Somit hatte ich das teuerste Eprom meines Lebens erworben.
Ausgestattet mit den, durch Einbau des neuen Zeichensatzes erworbenen Hardwarekenntnissen (ansonsten bin ich Hardware-Laie) ging ich daran, für meine Programme, in denen ich meist Umlaute, Kleinschrit und Grafiken anwende, eine eigene Lösung zu finden.
Es war einfacher als ich dachte und ich hätte viel Zeit und Geld sparen können, wenn die Firma Apple in Ihrem Handbuch die bewußte Trennstelle auf der Platine erwähnt hätte und auf die Möglichkeit hingewiesen hätte, den Zeichensatz durch ein Eprom zu ersetzen.
Mittlerweile kann ich je nach Bedarf aus 8 Zeichensätzen mit verschiedenen Schrift- und Grafikzeichen (= 4 übereinandergelötete Eproms) wählen, habe 4 und 2 an der richtigen Stelle, habe so ganz nebenbei Apple-Steno, Up-Scroll und ein editierfreundliches LIST implementiert.
Natürlich geht das alles nur über die 16-K-RAM-Karte, aber ich nehme an, daß viele Apple-User ihr Gerät auf 64 K aufgerüstet haben.
Ein Textfile steuert über die EXEC-Funktion die Lade- und Programmänderungsvorgänge. Die Programm-Listings befinden sich auf den folgenden Seiten.
Im Apple selbst sind folgende Änderungen vorzunehmen:
a) nach Entfernen des Gehäuses erkennt man am linken Rand der Platine eine Reihe Buchstaben (A - K) und am vorderen Rand eine Zahlenreihe (1 - 14).
Zwischen den sich daraus ergebenden Ko-ordinaten B6 und B7 befindet sich eine Trennstelle, die vorsichtig zu durchtrennen ist. An den dem IC B6 näherliegenden Teil (oder an dem hochzubiegenden Pin 19 des Zeichensatz-Eproms) wird ein Kabel angelötet, das mit dem mittleren Anschluß eines einpoligen Umschalters verbunden wird. Die beiden äußeren Anschlüsse dieses Schalters werden am Pin 1 (+5V) und am Pin 8 (Gnd) des Game I/O Connectors (siehe Apple Reference Manual auf Seite 100) angeschlossen. Mit diesem Umschalter können die zwei Zeichensätze im Eprom angewählt werden. Ein Eprom 2716, das einen neuen Zeichensatz und zwecks Kompatibilität den alten Zeichensatz enthalten sollte, ersetzt den alten Zeichengenerator auf A5.
b) Zum Spiele E/A Anschluß sind noch drei weitere Verbindungen herzustellen, was zweckmäßigerweise über einen separaten Stecker geschieht. Diese Anschlüsse verbinden das Keybord mit den Pushbutton-Kontakten, weil das Programm so ausgelegt ist, daß nach Abfrage der Flaginputs PB0 - PB2 entschieden wird, welcher Buchstabe, welches Zeichen oder gar ein vollständiger Befehl auszugeben ist.
Die Shift-Taste erhält Kontakt zum PB2, weil dies die Kompatibiltät mit der 80-Zeichenkarte erhält, während wir den PB0 mit der CTRL-Taste verbunden haben. Zwischen Pin 1 (+5V) und PB1 haben wir einen Schalter gelegt, und damit erreicht, daß sich Codes für Groß- und Kleinbuchstaben oder Großbuchstaben anstelle getippter Kleinbuchstaben nach Verlassen der KEYIN-Routine des Monitors im Akkumulator befinden. Es ist ratsam, einen Widerstand von 3,9 K zwischen PB1 und Gnd zu legen (A.U.G.E. User-Magazin 8/82 Seite 26).
Die Anschlüsse zur Shift- und Ctrl-Taste werden an Punkt 3 (Shift) und Punkt 24 (Ctrl) der kleinen Key-

board Encoder Platine angelötet. Man findet diese beiden Punkte leicht. Sie befinden sich oberhalb des Steckkontaktes, der Hauptkeyboardplatine und Encoderboard verbindet.
Programme:
ASCROLL.OBJ0,
NEW-KBD #3.OBJ0
und NEW-KBD #3.OBJ1 werden als Binärfiles, EM-NK wird als Textfile und Kleinschrift als Applesoft-File auf Diskette gespeichert.
Nach RUN Kleinschrift befindet sich der Apple im Kleinschrift-Modus. Man erkennt dies am schnell blinkenden Cursor, denn der Cursor steht im Normalmodus still. Dementsprechend funktioniert FLASH auch nicht mehr, das ich allerdings ohnehin nie benötigte.

LESER-SERVICE: Zeichensätze und Programme auf Diskette sind von unserer Service-Abteilung erhältlich. Mehr darüber am Heftende.

PEEK(	SAVE	INVERSE	HIMEM:	OPEN	RIGHT$(	STR$(	CHR$(	DATA	HOME	CALL-151		RESET
!	"	#	$	%	&	'	(	)	LIST	*	=	RESET
POKE	LOAD	NORMAL	LOMEM:	MID$(	LEFT$(	VAL(	ASC(	INPUT	RND(1)	CALL		RESET
1	2	3	4	5	6	7	8	9	0	:	-	RESET

ESC	(Q)	WRITE	EXEC	READ	TAB(	GOTO	ü	(U)	ö	§	REPT	RETURN
ESC	Q	W	E	R	T	Z	U	I	O	P	REPT	
ESC	(A)	(W)	(E)	(R)	(T)	(Z)	ü	(I)	ö	(P)	REPT	
ESC	q	w	e	r	t	z	u	i	o	p	REPT	Car.Ret

CTRL	Ä	B	DELETE	FOR	GOSUB	HTAB	THEN	STEP	(O)	HCOLOR=	HTAB	
CTRL	A	S	D	F	G	H	J	K	L	+	←	>>>>>>>
CTRL	ä	(S)	(D)	(F)	(G)	(H)	(J)	(K)	(L)	HPLOT	←	
CTRL	a	s	d	f	g	h	j	k	l	;	←	→

SHIFT	RUN	NEXT	CLOSE	VTAB	SPEED=	^	MAXFILES	DRAW			SHIFT
SHIFT	Y	X	C	V	B	N	M	<	>	?	SHIFT
SHIFT	(Y)	(X)	(C)	(V)	(B)	(N)	(M)	3D0G		FRE(0)	SHIFT
SHIFT	y	x	c	v	b	n	m	,	.	/	SHIFT

C A T A L O G
S P A C E
H o m e c o m p u t e r
S P A C E

```
SOURCE FILE: RSCROLL
0021:            1 WNDWDTH EQU  $21
0022:            2 WNDTOP  EQU  $22
0023:            3 WNDBTM  EQU  $23
0028:            4 BASL    EQU  $28
0029:            5 BASH    EQU  $29
002A:            6 BAS2L   EQU  $2A
002B:            7 BAS2H   EQU  $2B
C030:            8 SPKR    EQU  $C030
C080:            9 RD4K1   EQU  $C080
C081:           10 WR4K1   EQU  $C081
C082:           11 RAMOFF  EQU  $C082
C083:           12 RDWR4K1 EQU  $C083
C088:           13 RD4K2   EQU  $C088
C089:           14 WR4K2   EQU  $C089
C08B:           15 RDWR4K2 EQU  $C08B
D000:           16 RAMPKT0 EQU  $D000
D031:           17 RAMPKT1 EQU  $D031
FBDD:           18 BELL1.5 EQU  $FBDD
FC22:           19 VTAB    EQU  $FC22
FC24:           20 VTABZ   EQU  $FC24
FC9E:           21 CLEOLZ  EQU  $FC9E
FCA8:           22 WAIT    EQU  $FCA8
----- NEXT OBJECT FILE NAME IS RSCROLL.OBJ0
1000:           23         ORG  $1000
1000:AD 89 C0   24 INIT    LDA  WR4K2
1003:AD 89 C0   25         LDA  WR4K2
1006:A2 44      26         LDX  #68
1008:BD 25 10   27 INIT1   LDA  RSCROLL,X
100B:9D 00 D0   28         STA  RAMPKT0,X
100E:CA         29         DEX
100F:10 F7      30         BPL  INIT1
1011:A2 11      31         LDX  #17
1013:BD 69 10   32 INIT2   LDA  NBELL,X
1016:9D DD FB   33         STA  BELL1.5,X
1019:CA         34         DEX
101A:10 F7      35         BPL  INIT2
101C:A9 C6      36         LDA  #$C6
101E:8D 1F FC   37         STA  $FC1F
1021:AD 80 C0   38         LDA  RD4K1
1024:60         39         RTS
1025:A5 23      40 RSCROLL LDA  WNDBTM
1027:38         41         SEC
1028:E9 01      42         SBC  #1
102A:48         43         PHA
102B:20 24 FC   44         JSR  VTABZ
102E:A5 28      45 RSCRL1  LDA  BASL
1030:85 2A      46         STA  BAS2L
1032:A5 29      47         LDA  BASH
1034:85 2B      48         STA  BAS2H
1036:A4 21      49         LDY  WNDWDTH
1038:88         50         DEY
1039:68         51         PLA
103A:38         52         SEC
103B:E9 01      53         SBC  #1
103D:C5 22      54         CMP  WNDTOP
103F:30 0D      55         BMI  RSCRL3
1041:48         56         PHA
1042:20 24 FC   57         JSR  VTABZ
1045:B1 28      58 RSCRL2  LDA  (BASL),Y
1047:91 2A      59         STA  (BAS2L),Y
1049:88         60         DEY
104A:10 F9      61         BPL  RSCRL2
104C:30 E0      62         BMI  RSCRL1
104E:A0 00      63 RSCRL3  LDY  #0
1050:20 9E FC   64         JSR  CLEOLZ
1053:4C 22 FC   65         JMP  VTAB
1056:A9 40      66 NBELL1A LDA  #$40
1058:20 A8 FC   67         JSR  WAIT
105B:A0 C0      68         LDY  #$C0
105D:A9 0C      69 NBELL2  LDA  #$0C
105F:20 A8 FC   70         JSR  WAIT
1062:AD 30 C0   71         LDA  SPKR
1065:88         72         DEY
1066:D0 F5      73         BNE  NBELL2
1068:60         74         RTS
1069:AD 88 C0   75 NBELL   LDA  RD4K2
106C:20 31 D0   76         JSR  RAMPKT1
106F:4C EC FB   77         JMP  BELL1.5+15
1072:AD 88 C0   78 NEWUP   LDA  RD4K2
1075:20 00 D0   79         JSR  RAMPKT0
1078:AD 80 C0   80 NEWUP1  LDA  RD4K1

*** SUCCESSFUL ASSEMBLY: NO ERRORS
```

EX-NK

```
 1 CALL-151
 2 C081
 3 C081
 4 1000:A0 00 A2 00 84 3C 84 42 A9 D
   0 85 3D 85 43 A9 FF 85 3E 85 3F 4
   C 2C FE
 5 1000G
 6 FD82:EA EA
 7 D708:28
 8 D70F:0
 9 FBB3:8D 80 C0 20 84 FE 60
10 FA63:20 B3 FB
11 BLOADRSCROLL.OBJ0
12 1000G
13 C08B
14 C08B
15 BLOAD NEW-KBD #3.OBJ0
16 BLOAD NEW-KBD #3.OBJ1
17 C080
18 3D0G
```

Kleinschrift

```
10  HOME
55  PRINT "RSCROLL & BELL in Para
     llel-RAM laden undSub-Progra
     mme 'UP' und 'BELL1' ändern.
     "
70  PRINT : PRINT "Kurzschrift &
     Groß/Kleinschrift          ak
     tivieren"
80  PRINT : PRINT "40-Spalten-LIS
     T einführen"
100  PRINT  CHR$ (4);"EXECEX-NK"
```

```
§

SOURCE FILE: NEW-KBD #3
0021:            1 WNDWDTH EQU  $21
0024:            2 CH      EQU  $24
0028:            3 BASL    EQU  $28
004E:            4 RNDL    EQU  $4E
004F:            5 RNDH    EQU  $4F
0200:            6 IN      EQU  $200
C000:            7 KBD     EQU  $C000
C010:            8 KBDSTRB EQU  $C010
C061:            9 PB0     EQU  $C061
C062:           10 PB1     EQU  $C062
C063:           11 PB2     EQU  $C063
C080:           12 RD.12K  EQU  $C080
C081:           13 WRT.12K EQU  $C081
C088:           14 RD.4K   EQU  $C088
C089:           15 WRT.4K  EQU  $C089
C08B:           16 R.W.4K  EQU  $C08B
FDED:           17 COUT    EQU  $FDED
----- NEXT OBJECT FILE NAME IS NEW-KBD #3.OBJ0
D200:           18         ORG  $D200
D200:20 13 D2   19         JSR  NEWKYIN
D203:48         20         PHA
D204:AD 62 C0   21         LDA  PB1
D207:30 08      22         BMI  VORPGM1
D209:68         23         PLA
D20A:C9 E1      24         CMP  #$E1
D20C:90 02      25         BCC  VORPGM0
D20E:29 DF      26         AND  #$DF
D210:60         27 VORPGM0 RTS
D211:68         28 VORPGM1 PLA
D212:60         29         RTS
D213:48         30 NEWKYIN PHA
D214:E6 4E      31         INC  RNDL
D216:D0 0E      32         BNE  NEWKY1
D218:E6 4F      33         INC  RNDH
D21A:A5 4F      34         LDA  RNDH
D21C:29 1F      35         AND  #$1F
D21E:D0 06      36         BNE  NEWKY1
D220:B1 28      37         LDA  (BASL),Y
D222:49 80      38         EOR  #$80
D224:91 28      39         STA  (BASL),Y
D226:68         40 NEWKY1  PLA
D227:2C 00 C0   41         BIT  KBD
D22A:10 E7      42         BPL  NEWKYIN
D22C:91 28      43         STA  (BASL),Y
D22E:AD 00 C0   44         LDA  KBD
D231:2C 10 C0   45         BIT  KBDSTRB
D234:C9 D9      46         CMP  #$D9
D236:D0 04      47         BNE  NEWKY2
D238:A9 DA      48         LDA  #$DA
D23A:D0 16      49         BNE  NEWKY5
D23C:C9 DA      50 NEWKY2  CMP  #$DA
D23E:D0 04      51         BNE  NEWKY3
D240:A9 D9      52         LDA  #$D9
D242:D0 0E      53         BNE  NEWKY5
D244:C9 99      54 NEWKY3  CMP  #$99
D246:D0 04      55         BNE  NEWKY4
D248:A9 9A      56         LDA  #$9A
D24A:D0 06      57         BNE  NEWKY5
D24C:C9 9A      58 NEWKY4  CMP  #$9A
D24E:D0 02      59         BNE  NEWKY5
D250:A9 99      60         LDA  #$99
D252:48         61 NEWKY5  PHA
D253:AD 61 C0   62         LDA  PB0
D256:10 14      63         BPL  CTRL
D258:AD 63 C0   64         LDA  PB2
D25B:30 03      65         BMI  NEWKY6
D25D:4C 70 D3   66         JMP  SHIFT
D260:68         67 NEWKY6  PLA
D261:C9 C1      68         CMP  #$C1
D263:90 06      69         BCC  NEWKY7
D265:C9 DB      70         CMP  #$DB
D267:B0 02      71         BCS  NEWKY7
D269:09 20      72         ORA  #$20
D26B:60         73 NEWKY7  RTS
D26C:AD 63 C0   74 CTRL    LDA  PB2
D26F:30 03      75         BMI  CTRL0
D271:4C AA D3   76         JMP  SHFT.C
D274:68         77 CTRL0   PLA
D275:C9 81      78         CMP  #$81
D277:D0 03      79         BNE  CTRL1
D279:A9 FB      80         LDA  #$FB
D27B:60         81         RTS
D27C:C9 8F      82 CTRL1   CMP  #$8F
D27E:D0 03      83         BNE  CTRL2
D280:A9 FC      84         LDA  #$FC
D282:60         85         RTS
D283:C9 95      86 CTRL2   CMP  #$95
D285:D0 03      87         BNE  CTRL3
D287:A9 FD      88         LDA  #$FD
D289:60         89         RTS
D28A:C9 91      90 CTRL3   CMP  #$91
D28C:D0 03      91         BNE  CTRL4
D28E:A9 81      92         LDA  #$81
D290:60         93         RTS
D291:C9 AD      94 CTRL4   CMP  #$AD
D293:D0 03      95         BNE  CTRL5
D295:A9 DF      96         LDA  #$DF
D297:60         97         RTS
D298:C9 B1      98 CTRL5   CMP  #$B1
D29A:D0 08      99         BNE  CTRL6
D29C:20 8C D5  100         JSR  PRT.IN
D29F:50 4F 4B  101         DCI  'POKE'
D2A2:C5
D2A3:00        102         BRK
D2A4:C9 B2     103 CTRL6   CMP  #$B2
D2A6:D0 08     104         BNE  CTRL7
D2A8:20 8C D5  105         JSR  PRT.IN
D2AB:4C 4F 41  106         DCI  'LOAD'
D2AE:C4
D2AF:00        107         BRK
D2B0:C9 B3     108 CTRL7   CMP  #$B3
D2B2:D0 0A     109         BNE  CTRL8
D2B4:20 8C D5  110         JSR  PRT.IN
D2B7:4E 4F 52  111         DCI  'NORMAL'
D2BA:4D 41 CC
D2BD:00        112         BRK
D2BE:C9 B4     113 CTRL8   CMP  #$B4
D2C0:D0 0A     114         BNE  CTRL9
D2C2:20 8C D5  115         JSR  PRT.IN
D2C5:4C 4F 4D  116         DCI  'LOMEM:'
D2C8:45 4D BA
D2CB:00        117         BRK
D2CC:C9 B5     118 CTRL9   CMP  #$B5
D2CE:D0 09     119         BNE  CTRL10
D2D0:20 8C D5  120         JSR  PRT.IN
D2D3:4D 49 44  121         DCI  'MID$('
D2D6:24 A8
D2D8:00        122         BRK
D2D9:C9 B6     123 CTRL10  CMP  #$B6
```

```
D2DB:D0 0A      124          BNE  CTRL11
D2DD:20 8C D5   125          JSR  PRT.IN
D2E0:4C 45 46   126          DCI  'LEFT$('
D2E3:54 24 A8
D2E6:00         127          BRK
D2E7:C9 B7      128 CTRL11   CMP  #$B7
D2E9:D0 08      129          BNE  CTRL12
D2EB:20 8C D5   130          JSR  PRT.IN
D2EE:56 41 4C   131          DCI  'VAL('
D2F1:A8
D2F2:00         132          BRK
D2F3:C9 B8      133 CTRL12   CMP  #$B8
D2F5:D0 08      134          BNE  CTRL13
D2F7:20 8C D5   135          JSR  PRT.IN
D2FA:41 53 43   136          DCI  'ASC('
D2FD:A8
D2FE:00         137          BRK
D2FF:C9 B9      138 CTRL13   CMP  #$B9
D301:D0 09      139          BNE  CTRL14
D303:20 8C D5   140          JSR  PRT.IN
D306:49 4E 50   141          DCI  'INPUT'
D309:55 D4
D30B:00         142          BRK
D30C:C9 B0      143 CTRL14   CMP  #$B0
D30E:D0 0A      144          BNE  CTRL15
D310:20 8C D5   145          JSR  PRT.IN
D313:52 4E 44   146          DCI  'RND(1)'
D316:28 31 A9
D319:00         147          BRK
D31A:C9 BA      148 CTRL15   CMP  #$BA
D31C:D0 08      149          BNE  CTRL16
D31E:20 8C D5   150          JSR  PRT.IN
D321:43 41 4C   151          DCI  'CALL'
D324:CC
D325:00         152          BRK
D326:C9 BB      153 CTRL16   CMP  #$BB
D328:D0 09      154          BNE  CTRL17
D32A:20 8C D5   155          JSR  PRT.IN
D32D:48 50 4C   156          DCI  'HPLOT'
D330:4F D4
D332:00         157          BRK
D333:C9 AC      158 CTRL17   CMP  #$AC
D335:D0 08      159          BNE  CTRL18
D337:20 8C D5   160          JSR  PRT.IN
D33A:33 44 30   161          DCI  '3D0G'
D33D:C7
D33E:00         162          BRK
D33F:C9 AE      163 CTRL18   CMP  #$AE
D341:D0 0B      164          BNE  CTRL19
D343:20 8C D5   165          JSR  PRT.IN
D346:52 45 53   166          DCI  'RESTORE'
D349:54 4F 52
D34C:C5
D34D:00         167          BRK
D34E:C9 A0      168 CTRL19   CMP  #$A0
D350:D0 10      169          BNE  CTRL20
D352:20 8C D5   170          JSR  PRT.IN
D355:48 6F 6D   171          DCI  'Homecomputer'
D358:65 63 6F
D35B:6D 70 75
D35E:74 65 F2
D361:00         172          BRK
D362:C9 AF      173 CTRL20   CMP  #$AF
D364:D0 09      174          BNE  CTRL21
D366:20 8C D5   175          JSR  PRT.IN
D369:46 52 45   176          DCI  'FRE(0)'
D36C:28 30 A9
D36F:60         177 CTRL21   RTS
D370:68         178 SHIFT    PLA
D371:C9 DD      179          CMP  #$DD
D373:D0 03      180          BNE  SHIFT1
D375:A9 CD      181          LDA  #$CD
D377:60         182          RTS
D378:C9 DE      183 SHIFT1   CMP  #$DE
D37A:D0 03      184          BNE  SHIFT2
D37C:A9 CE      185          LDA  #$CE
D37E:60         186          RTS
D37F:C9 C0      187 SHIFT2   CMP  #$C0
D381:D0 03      188          BNE  SHIFT3
D383:A9 D0      189          LDA  #$D0
D385:60         190          RTS
D386:C9 B0      191 SHIFT3   CMP  #$B0
D388:D0 08      192          BNE  SHIFT4
D38A:20 8C D5   193          JSR  PRT.IN
D38D:4C 49 53   194          DCI  'LIST'
D390:D4
D391:00         195          BRK
D392:C9 95      196 SHIFT4   CMP  #$95
D394:D0 13      197          BNE  SHIFT5
D396:A4 24      198 SHIFT4A  LDY  CH
D398:B1 28      199          LDA  (BASL),Y
D39A:C9 A0      200          CMP  #$A0
D39C:D0 01      201          BNE  SHIFT4B
D39E:60         202          RTS
D39F:20 ED FD   203 SHIFT4B  JSR  COUT
D3A2:9D 00 02   204          STA  IN,X
D3A5:E8         205          INX
D3A6:4C 96 D3   206          JMP  SHIFT4A
D3A9:60         207 SHIFT5   RTS
D3AA:68         208 SHFT.C   PLA
D3AB:C9 80      209          CMP  #$80
D3AD:D0 03      210          BNE  SHFT.C0
D3AF:A9 C0      211          LDA  #$C0
D3B1:60         212          RTS
D3B2:C9 9E      213 SHFT.C0  CMP  #$9E
D3B4:D0 03      214          BNE  SHFT.C1
D3B6:A9 DE      215          LDA  #$DE
D3B8:60         216          RTS
D3B9:C9 81      217 SHFT.C1  CMP  #$81
D3BB:D0 03      218          BNE  SHFT.C2
D3BD:A9 DB      219          LDA  #$DB
D3BF:60         220          RTS
D3C0:C9 8F      221 SHFT.C2  CMP  #$8F
D3C2:D0 03      222          BNE  SHFT.C3
D3C4:A9 DC      223          LDA  #$DC
D3C6:60         224          RTS
D3C7:C9 95      225 SHFT.C3  CMP  #$95
D3C9:D0 03      226          BNE  SHFT.C4
D3CB:A9 DD      227          LDA  #$DD
D3CD:60         228          RTS
D3CE:C9 93      229 SHFT.C4  CMP  #$93
D3D0:D0 03      230          BNE  SHFT.C5
D3D2:A9 FE      231          LDA  #$FE
D3D4:60         232          RTS
D3D5:C9 BF      233 SHFT.C5  CMP  #$BF
D3D7:D0 03      234          BNE  SHFT.C6
D3D9:A9 FF      235          LDA  #$FF
D3DB:60         236          RTS
D3DC:C9 A0      237 SHFT.C6  CMP  #$A0
D3DE:D0 0B      238          BNE  SHFT.C8
D3E0:20 8C D5   239          JSR  PRT.IN
D3E3:43 41 54   240          DCI  'CATALOG'
D3E6:41 4C 4F
D3E9:C7
D3EA:00         241          BRK
D3EB:C9 8D      242 SHFT.C8  CMP  #$8D
D3ED:D0 0A      243          BNE  SHFT.C9
D3EF:20 8C D5   244          JSR  PRT.IN
D3F2:52 45 54   245          DCI  'RETURN'
D3F5:55 52 CE
D3F8:00         246          BRK
D3F9:C9 A1      247 SHFT.C9  CMP  #$A1
D3FB:D0 09      248          BNE  SHFT.C10
D3FD:20 8C D5   249          JSR  PRT.IN
D400:50 45 45   250          DCI  'PEEK('
D403:4B A8
D405:00         251          BRK
D406:C9 A2      252 SHFT.C10 CMP  #$A2
D408:D0 08      253          BNE  SHFT.C11
D40A:20 8C D5   254          JSR  PRT.IN
D40D:53 41 56   255          DCI  'SAVE'
D410:C5
D411:00         256          BRK
D412:C9 A3      257 SHFT.C11 CMP  #$A3
D414:D0 0B      258          BNE  SHFT.C12
D416:20 8C D5   259          JSR  PRT.IN
D419:49 4E 56   260          DCI  'INVERSE'
D41C:45 52 53
D41F:C5
D420:00         261          BRK
D421:C9 A4      262 SHFT.C12 CMP  #$A4
D423:D0 0A      263          BNE  SHFT.C13
D425:20 8C D5   264          JSR  PRT.IN
D428:48 49 4D   265          DCI  'HIMEM:'
D42B:45 4D BA
D42E:00         266          BRK
D42F:C9 A5      267 SHFT.C13 CMP  #$A5
D431:D0 08      268          BNE  SHFT.C14
D433:20 8C D5   269          JSR  PRT.IN
D436:4F 50 45   270          DCI  'OPEN'
D439:CE
D43A:00         271          BRK
D43B:C9 A6      272 SHFT.C14 CMP  #$A6
D43D:D0 0B      273          BNE  SHFT.C15
D43F:20 8C D5   274          JSR  PRT.IN
D442:52 49 47   275          DCI  'RIGHT$('
D445:48 54 24
D448:A8
D449:00         276          BRK
D44A:C9 A7      277 SHFT.C15 CMP  #$A7
D44C:D0 09      278          BNE  SHFT.C16
D44E:20 8C D5   279          JSR  PRT.IN
D451:53 54 52   280          DCI  'STR$('
D454:24 A8
D456:00         281          BRK
D457:C9 A8      282 SHFT.C16 CMP  #$A8
D459:D0 09      283          BNE  SHFT.C17
D45B:20 8C D5   284          JSR  PRT.IN
D45E:43 48 52   285          DCI  'CHR$('
D461:24 A8
D463:00         286          BRK
D464:C9 A9      287 SHFT.C17 CMP  #$A9
D466:D0 08      288          BNE  SHFT.C18
D468:20 8C D5   289          JSR  PRT.IN
D46B:44 41 54   290          DCI  'DATA'
D46E:C1
D46F:00         291          BRK
D470:C9 B0      292 SHFT.C18 CMP  #$B0
D472:D0 08      293          BNE  SHFT.C19
D474:20 8C D5   294          JSR  PRT.IN
D477:48 4F 4D   295          DCI  'HOME'
D47A:C5
D47B:00         296          BRK
D47C:C9 AA      297 SHFT.C19 CMP  #$AA
D47E:D0 0C      298          BNE  SHFT.C20
D480:20 8C D5   299          JSR  PRT.IN
D483:43 41 4C   300          DCI  'CALL-151'
D486:4C 2D 31
D489:35 B1
D48B:00         301          BRK
D48C:C9 BD      302 SHFT.C20 CMP  #$BD
D48E:D0 0D      303          BNE  SHFT.C21
D490:20 8C D5   304          JSR  PRT.IN
D493:50 4F 4B   305          DCI  'POKE1657,'
D496:45 31 36
D499:35 37 AC
D49C:00         306          BRK
D49D:C9 97      307 SHFT.C21 CMP  #$97
D49F:D0 09      308          BNE  SHFT.C22
D4A1:20 8C D5   309          JSR  PRT.IN
D4A4:57 52 49   310          DCI  'WRITE'
D4A7:54 C5
D4A9:00         311          BRK
D4AA:C9 85      312 SHFT.C22 CMP  #$85
D4AC:D0 08      313          BNE  SHFT.C23
D4AE:20 8C D5   314          JSR  PRT.IN
D4B1:45 58 45   315          DCI  'EXEC'
D4B4:C3
D4B5:00         316          BRK
D4B6:C9 92      317 SHFT.C23 CMP  #$92
D4B8:D0 08      318          BNE  SHFT.C24
D4BA:20 8C D5   319          JSR  PRT.IN
D4BD:52 45 41   320          DCI  'READ'
D4C0:C4
D4C1:00         321          BRK
D4C2:C9 94      322 SHFT.C24 CMP  #$94
D4C4:D0 08      323          BNE  SHFT.C25
D4C6:20 8C D5   324          JSR  PRT.IN
D4C9:54 41 42   325          DCI  'TAB('
D4CC:A8
D4CD:00         326          BRK
D4CE:C9 9A      327 SHFT.C25 CMP  #$9A
D4D0:D0 08      328          BNE  SHFT.C26
D4D2:20 8C D5   329          JSR  PRT.IN
D4D5:47 4F 54   330          DCI  'GOTO'
D4D8:CF
D4D9:00         331          BRK
D4DA:C9 84      332 SHFT.C26 CMP  #$84
D4DC:D0 0A      333          BNE  SHFT.C27
D4DE:20 8C D5   334          JSR  PRT.IN
D4E1:44 45 4C   335          DCI  'DELETE'
D4E4:45 54 C5
D4E7:00         336          BRK
D4E8:C9 86      337 SHFT.C27 CMP  #$86
D4EA:D0 07      338          BNE  SHFT.C28
D4EC:20 8C D5   339          JSR  PRT.IN
D4EF:46 4F D2   340          DCI  'FOR'
D4F2:00         341          BRK
D4F3:C9 87      342 SHFT.C28 CMP  #$87
D4F5:D0 09      343          BNE  SHFT.C29
D4F7:20 8C D5   344          JSR  PRT.IN
D4FA:47 4F 53   345          DCI  'GOSUB'
D4FD:55 C2
D4FF:00         346          BRK
```

```
D500:C9 88     347 SHFT.C29 CMP #$88
D502:D0 08     348          BNE SHFT.C30
D504:20 8C D5  349          JSR PRT.IN
D507:48 54 41  350          DCI 'HTAB'
D50A:C2
D50B:00        351          BRK
D50C:C9 8A     352 SHFT.C30 CMP #$8A
D50E:D0 08     353          BNE SHFT.C31
D510:20 8C D5  354          JSR PRT.IN
D513:54 48 45  355          DCI 'THEN'
D516:CE
D517:00        356          BRK
D518:C9 8B     357 SHFT.C31 CMP #$8B
D51A:D0 08     358          BNE SHFT.C32
D51C:20 8C D5  359          JSR PRT.IN
D51F:53 54 45  360          DCI 'STEP'
D522:D0
D523:00        361          BRK
D524:C9 AB     362 SHFT.C32 CMP #$AB
D526:D0 0B     363          BNE SHFT.C33
D528:20 8C D5  364          JSR PRT.IN
D52B:48 43 4F  365          DCI 'HCOLOR='
D52E:4C 4F 52
D531:BD
D532:00        366          BRK
D533:C9 99     367 SHFT.C33 CMP #$99
D535:D0 07     368          BNE SHFT.C34
D537:20 8C D5  369          JSR PRT.IN
D53A:52 55 CE  370          DCI 'RUN'
D53D:00        371          BRK
D53E:C9 98     372 SHFT.C34 CMP #$98
D540:D0 08     373          BNE SHFT.C35
D542:20 8C D5  374          JSR PRT.IN
D545:4E 45 58  375          DCI 'NEXT'
D548:D4
D549:00        376          BRK
D54A:C9 83     377 SHFT.C35 CMP #$83
D54C:D0 09     378          BNE SHFT.C36
D54E:20 8C D5  379          JSR PRT.IN
D551:43 4C 4F  380          DCI 'CLOSE'
D554:53 C5
D556:00        381          BRK
D557:C9 96     382 SHFT.C36 CMP #$96
D559:D0 08     383          BNE SHFT.C37
D55B:20 8C D5  384          JSR PRT.IN
D55E:56 54 41  385          DCI 'VTAB'
D561:C2
D562:00        386          BRK
D563:C9 82     387 SHFT.C37 CMP #$82
D565:D0 0A     388          BNE SHFT.C38
D567:20 8C D5  389          JSR PRT.IN
D56A:53 50 45  390          DCI 'SPEED='
D56D:45 44 BD
D570:00        391          BRK
D571:C9 BC     392 SHFT.C38 CMP #$BC
D573:D0 07     393          BNE SHFT.C39
D575:20 8C D5  394          JSR PRT.IN
D578:44 52 41  395          DCI 'DRAW'
D57B:D7
D57C:C9 9D     396 SHFT.C39 CMP #$9D
D57E:D0 0B     397          BNE SHFT.C40
D580:20 8C D5  398          JSR PRT.IN
D583:4D 41 58  399          DCI 'MAXFILES'
D586:46 49 4C
D589:45 D3
D58B:60        400 SHFT.C40 RTS
D58C:68        401 PRT.IN   PLA
D58D:8D 98 D5  402          STA PRT.IN2+1
D590:68        403          PLA
D591:8D 99 D5  404          STA PRT.IN2+2
D594:20 A8 D5  405 PRT.IN1  JSR PRT.IN3
D597:AD FF FF  406 PRT.IN2  LDA $FFFF
D59A:30 15     407          BMI PRT.IN5
D59C:09 80     408          ORA #$80
D59E:20 ED FD  409          JSR COUT
D5A1:9D 00 02  410          STA IN,X
D5A4:E8        411          INX
D5A5:4C 94 D5  412          JMP PRT.IN1
D5A8:EE 98 D5  413 PRT.IN3  INC PRT.IN2+1
D5AB:D0 03     414          BNE PRT.IN4
D5AD:EE 99 D5  415          INC PRT.IN2+2
D5B0:60        416 PRT.IN4  RTS
D5B1:60        417 PRT.IN5  RTS
----- NEXT OBJECT FILE NAME IS NEW-KBD #3.OBJ1
FD1B:          418          ORG $FD1B
FD1B:2C 8B C0  419 KEYIN    BIT $C08B
FD1E:2C 8B C0  420          BIT $C08B
FD21:EA        421          NOP
FD22:EA        422          NOP
FD23:EA        423          NOP
FD24:20 00 D2  424          JSR $D200
FD27:2C 80 C0  425          BIT $C080
FD2A:2C 80 C0  426          BIT $C080
FD2D:EA        427          NOP
FD2E:60        428          RTS

*** SUCCESSFUL ASSEMBLY: NO ERRORS
```

```
ÜLIST

10  PRINT  CHR$ (4);"MAXFILES2"
11  HIMEM: 8192
12 RW = 34083:TR = 34097:SE = 340
     98:CD = 34105:VB = 33792:CB =
     35072:D$ =  CHR$ (4)
13  ONERR  GOTO 200
20 ADR = 34048: FOR I = 0 TO 12 STEP
     3: IF  PEEK (ADR + I) = 76 THEN
      NEXT : GOTO 50
25  PRINT D$;"BLOADDISKOS.OBJ0"
50  CALL AD + 3
55  HOME
60  CALL 34114:FZ =  PEEK (34123)
     : CALL AD + 9
65 PV =  INT (FZ / 16) + 1
70  CALL AD
100  HOME : GOSUB 1000: GOSUB 102
     : GOTO 120
102  INVERSE : FOR I = 0 TO 15:H =
     I + 48: IF H > 57 THEN H = H
      + 7
104  VTAB 7 + I: PRINT  TAB( 2); CHR$
     (H); TAB( 40); CHR$ (H);: NEXT
     : POKE 2039,32
106  PRINT  TAB( 40);"    0123456
     789ABCDEF0123456789ABCDEF012
     ";: NORMAL
108  CALL AD + 12: RETURN
120  GOSUB 1100: GOSUB 1200:PG =
     0: GOSUB 1600
200  IF  PEEK (49152) < 128 THEN
     200
202  POKE 33,40: POKE 34,0: VTAB
     7
205  GET A$
210  IF A$ = "V" THEN 300
212  IF A$ = "L" THEN 440
214  IF A$ = "R" THEN 450
216  IF A$ = " " THEN 310
218  IF A$ = "K" THEN 400
220  IF A$ = "U" THEN 410
222  IF A$ = "F" THEN 460
224  IF A$ = "N" THEN 430
226  IF A$ = "B" THEN 520
228  IF A$ = "D" THEN 420
230  IF A$ = "G" THEN 100
232  IF A$ = "W" THEN 50
234  IF A$ = "E" THEN 500
236  IF A$ = "C" THEN 510
238  IF A$ = "S" THEN 570
240  IF A$ = "P" THEN 640
242  IF A$ = "=" THEN 600
244  IF A$ = "-" THEN 630
246  GOTO 200
300  GOSUB 1300: VTAB 2: HTAB 1: INPUT
     "NEW VOLUME NUMBER ? ";VN: IF
     VN < 0 OR VN > 254 THEN 300
302  CALL AD: POKE VB + 6,VN: CALL
     AD + 6
304  GOSUB 1400: GOTO 200
310  GOSUB 1500
311 CE = PG * 16:E = 1:PG = PG +
     1: IF PG > PV THEN PG = 0: GOTO
     311
312  GOSUB 1600: VTAB 8: HTAB 1
315  POKE 34117,CE: CALL 34126: GOTO
     200
400  IF PG = 0 THEN 200
401  GOSUB 1300: VTAB 2: HTAB 1: PRINT
     "LOCK ";: GET A$:A =  ASC (A
     $) - 65: IF A < 0 OR (A > 15
      AND A >  < 23) THEN 401
402  IF A = 23 THEN 850
403  GOSUB 900
406  VTAB 2: HTAB 6: PRINT NA$
407  PRINT D$;"LOCK"NA$
409  GOTO 50
410  IF PG = 0 THEN 200
411  GOSUB 1300: VTAB 2: HTAB 1: PRINT
     "UNLOCK ";: GET A$:A =  ASC
     (A$) - 65: IF A < 0 OR (A >
     15 AND A >  < 23) THEN 411
412  IF A = 23 THEN 800
413  GOSUB 900
416  VTAB 2: HTAB 8: PRINT NA$
417  PRINT D$;"UNLOCK"NA$
419  GOTO 50
420  IF PG = 0 THEN 200
421  GOSUB 1300: VTAB 2: HTAB 1: PRINT
     "DELETE ";: GET A$:A =  ASC
     (A$) - 65: IF A < 0 OR A > 1
     5 THEN 421
422  GOSUB 900
426  PRINT NA$;"   Y/N ? ";: GET
     A$: IF A$ = "N" THEN 200
427  PRINT : IF A$ >  < "Y" THEN
     421
428  PRINT D$;"DELETE"NA$
429  GOTO 50
430  IF PG = 0 THEN 200
431  GOSUB 1300: VTAB 2: HTAB 1: PRINT
     "RENAME ";: GET A$:A =  ASC
     (A$) - 65: IF A < 0 OR A > 1
     5 THEN 431
432  GOSUB 900
436  PRINT NA$;: INPUT "        ,?
     ";N$
438  PRINT D$;"RENAME"NA$","N$
439  GOTO 50
440  IF PG = 0 THEN 200
441  GOSUB 1300: VTAB 2: HTAB 1: PRINT
     "LOAD ";: GET A$:A =  ASC (A
     $) - 65: IF A < 0 OR A > 15 THEN
     441
442  GOSUB 900
443  PRINT NA$;"       Y/N ? ";: GET
     A$: IF A$ = "N" THEN 50
444  PRINT : IF A$ >  < "Y" THEN
     421
445  PRINT  CHR$ (4);"MAXFILES3"
446  HOME : IF P > 127 THEN P = P
      - 128
447  IF P = 0 THEN  RUN
448  IF P < 3 THEN  PRINT  CHR$ (
     4);"LOAD"NA$
449  IF P = 4 THEN  PRINT  CHR$ (
     4);"BLOAD"NA$
450  IF PG = 0 THEN 200
451  GOSUB 1300: VTAB 2: HTAB 1: PRINT
     "RUN ";: GET A$:A =  ASC (A$
     ) - 65: IF A < 0 OR A > 15 THEN
     451
452  GOSUB 900
453  PRINT NA$;"       Y/N ? ";: GET
     A$: IF A$ = "N" THEN 50
454  PRINT : IF A$ >  < "Y" THEN
     451
455  PRINT  CHR$ (4);"MAXFILES3"
456  HOME : IF P > 127 THEN P = P
      - 128
457  IF P = 0 THEN  RUN
458  IF P < 3 THEN  PRINT  CHR$ (
     4);"RUN"NA$
459  IF P = 4 THEN  PRINT  CHR$ (
     4);"BRUN"NA$
460  IF  PEEK (VB + 6) = 1 OR PG >
     0 THEN 200
461  GOSUB 1300: VTAB 2: PRINT "D
     ELETE DOS-TRACKS!
                   Y/N ? ";: GET A$:
      IF A$ = "N" THEN 50
462  IF A$ >  < "Y" THEN 461
463  FOR I = 56 TO 64 STEP 4: POKE
     VB + I,255: POKE VB + I + 1,
     255: NEXT : POKE VB + 57,254

464  POKE VB + 5,255
466  CALL AD + 6: GOTO 100
500  HOME
502  PRINT : PRINT D$"FP"
510  GOSUB 700
512  GET A$: IF  ASC (A$) <  > 13
      THEN 512
513  PRINT
514  PRINT D$"BRUNFID"
520  GOSUB 700
522  GET A$: IF  ASC (A$) <  > 13
      THEN 522
523  PRINT
524  PRINT D$"RUNCOPYA"
570  GOSUB 700
572  GET A$: IF  ASC (A$) <  > 13
      THEN 572
573  PRINT :ZB = 32: POKE CD,1: GOSUB
     590
574  GOSUB 1300: VTAB 2: HTAB 1: PRINT
     "INSERT DESTINATION DISK AND
      PRESS ";: INVERSE : PRINT "
     RETURN";: NORMAL
575  GET A$: IF  ASC (A$) <  > 13
      THEN 575
576  CALL AD: GOSUB 102: GOSUB 13
     00: VTAB 2: HTAB 1: PRINT "T
     RACKS 0 - 2 WILL NOW GET FIL
     LED  WITH  DOS-PROGRAM! Y/N
     ? ";
577  GET A$: IF A$ >  < "Y" THEN
     100
578  PRINT :ZB = 32: POKE CD,2: GOSUB
     590: CALL AD
579  FOR I = 56 TO 64 STEP 4: POKE
     VB + I,0: POKE VB + I + 1,0:
      NEXT : POKE VB + 5,0: CALL
     AD + 6
580  GOTO 50
590  FOR I = 0 TO 2: FOR J = 0 TO
     15
592  POKE TR,I: POKE SE,J: POKE S
     E + 4,ZB: CALL RW:ZB = ZB +
     1
594  NEXT : NEXT : RETURN
600  PRINT : GOSUB 650: PR# 1: PRINT
     CHR$ (27); CHR$ (19); CHR$
     (27); CHR$ (26);
601 I = 6
602 L1$ = "": FOR J = 0 TO 39:DZ =
     SCRN( J,I) +  SCRN( J,I + 1
     ) * 16
604  IF DZ > 159 THEN L1$ = L1$ +
     CHR$ (DZ - 128): GOTO 610
606  IF DZ < 32 THEN L1$ = L1$ +
     CHR$ (DZ + 64): GOTO 610
608 L1$ = L1$ +  CHR$ (DZ)
610  NEXT : IF  SCRN( 11,8) +  SCRN(
     11,9) * 16 = 48 THEN 620
612  IF I > 14 AND  SCRN( 7,I) +
     SCRN( 7,I + 1) * 16 = 160 THEN
     PR# 0: GOTO 200
620  VTAB 24: PRINT L1$;"

     "
622 I = I + 2: IF I < 48 THEN 602

624  PR# 0: GOTO 200
630  GOSUB 650: PR# 1:I = 14: GOTO
     602
640  PRINT : PR# 1: PRINT : PRINT
     : PRINT : PR# 0: GOTO 200
650  POKE 33,1: POKE 34,23: RETURN

700  GOSUB 1300: VTAB 2: HTAB 1: PRINT
     "DOS 3.3 SYSTEM MASTER DISKE
     TTE -> DRIVE1THEN PRESS ";: INVERSE
     : PRINT "RETURN";: NORMAL : RETURN
800  FOR I = 0 TO FZ:CE = I: GOSUB
     1700: IF P > 127 THEN  POKE
     CN,P - 128
810  NEXT : CALL AD + 9
820  GOTO 50
850  FOR I = 0 TO FZ:CE = I: GOSUB
     1700: IF P < 128 THEN  POKE
     CN,P + 128
860  NEXT : CALL AD + 9
870  GOTO 50
900 CE = (PG - 1) * 16 + A: GOSUB
     1700:NA$ = "": FOR I = 1 TO
     30:NA$ = NA$ +  CHR$ ( PEEK
     (CN + I)): NEXT : RETURN
999  END
1000  VTAB 1: HTAB 1: REM TITEL
1010  INVERSE : PRINT "-------- D
     OS 3.3  DISK-HANDLING ------
     --";
1015  NORMAL : PRINT "VOL DOS LOA
     D RUN  NXTPG LOCK UNLOCK FUL
     LRENAME BCKUP COPY DEL PRT G
     RAPH NEW EXIT---------------
     ------------------------VOL
      000 PG . OF . SECT-USED 000
      LEFT 000-------------------
     --------------------"
```

```
1020  INVERSE : VTAB 2: HTAB 1: PRINT
      "V";: HTAB 7: PRINT "S";: HTAB
      9: PRINT "L";: HTAB 14: PRINT
      "R";: HTAB 18: PRINT " ";: HTAB
      28: PRINT "K";: HTAB 30: PRINT
      "U";: HTAB 37: PRINT "F"
1025  HTAB 3: PRINT "N";: HTAB 8:
      PRINT "B";: HTAB 14: PRINT
      "C";: HTAB 19: PRINT "D";: HTAB
      23: PRINT "P";: HTAB 27: PRINT
      "G";: HTAB 35: PRINT "W";: HTAB
      37: PRINT "E"
1030  VTAB 5: HTAB 5: PRINT "000"
      ;: HTAB 12: PRINT "0";: HTAB
      17: PRINT "0";: HTAB 29: PRINT
      "000";: HTAB 38: PRINT "000"

1035  NORMAL : RETURN
1100  REM DISPLAY VOL.NR
1110 VN =  PEEK (VB + 6): VTAB 5:
      HTAB 5: POKE 68,VN: POKE 69
      ,0: INVERSE : CALL 44610: NORMAL
      : RETURN
1200  REM DISPLAY SECT-USED/LEFT
1210 FR =  PEEK (AD + 15) +  PEEK
      (AD + 16) * 256:US = 560 - F
      R: INVERSE
1215  VTAB 5: IF US >  < 0 THEN  HTAB
      31 -  INT ( LOG (US) /  LOG
      (10)): INVERSE : PRINT US
1220  VTAB 5: IF FR >  < 0 THEN  HTAB
      40 -  INT ( LOG (FR) /  LOG
      (10)): PRINT FR
1230  NORMAL : RETURN
1300  VTAB 2: HTAB 1: FOR I = 1 TO
      20: PRINT "     ";: NEXT : PRINT
      : RETURN
1400  GOSUB 1000: GOSUB 1100: GOSUB
      1200: GOSUB 1600: RETURN
1500  POKE 34,6: HOME : POKE 34,0
      : RETURN
1600  VTAB 5: HTAB 12: INVERSE : PRINT
      PG;: HTAB 17: PRINT PV: NORMAL
      : RETURN
1700 CH =  INT (CE / 7):CL = CE -
      CH * 7
1705 CN = CB + CH * 256 + 13 + CL
      * 35
1710 P =  PEEK (CN): RETURN
```

```
--------- DOS 3.3  DISK-HANDLING ---------
VOL DOS LOAD RUN  NXTPG LOCK UNLOCK FULL
RENAME BACKUP COPY DELETE GRAPH NEW EXIT
-----------------------------------------
VOL 001 PG 0 OF 2 SECT-USED 558 LEFT 002
-----------------------------------------
 0 ###*******.************************ 0
 1 ###*******.************************ 1
 2 ################################### 2
 3 ###******************************** 3
 4 ###******************************** 4
 5 ###******************************** 5
 6 ###******************************** 6
 7 ###******************************** 7
 8 ###******************************** 8
 9 ###******************************** 9
 A ###******************************** A
 B ###******************************** B
 C ###******************************** C
 D ###******************************** D
 E ###******************************** E
 F ###******************************** F

   0123456789ABCDEF0123456789ABCDEF012
```

```
-----------------------------------------
VOL 001 PG 1 OF 2 SECT-USED 558 LEFT 002
-----------------------------------------

*A 002 A HELLO
*I 018 B ANIMALS
*T 003 C APPLE PROMS
*I 006 D APPLESOFT
*I 026 E APPLEVISION
*I 017 F BIORHYTHM
*B 010 G BOOT13
*A 006 H BRIAN'S THEME
*B 003 I CHAIN
*I 009 J COLOR DEMO
*A 009 K COLOR DEMOSOFT
*I 009 L COPY
*B 003 M COPY.OBJO
*A 009 N COPYA
*A 019 O DISK-HANDLING
*T 019 P DISKOS
*B 005 A DISKOS.OBJO
*A 010 B EXEC DEMO
*B 020 C FID
*B 050 D FPBASIC
*B 002 E HDC
*A 006 F HELLO (DISK-ORIGINAL)
*B 050 G INTBASIC
*A 028 H LITTLE BRICK OUT
*A 003 I MAKE TEXT
*B 009 J MASTER CREATE
*B 027 K MUFFIN
*A 051 L PHONE LIST
*A 010 M RANDOM
*A 013 N RENUMBER
*A 039 O RENUMBER INSTRUCTIONS
*A 003 P RETRIEVE TEXT
```

```
-----------------------------------------
VOL 005 PG 0 OF 2 SECT-USED 285 LEFT 275
-----------------------------------------
 0 ...***..*........*........**.***..* 0
 1 ...***..*........*........**.***..* 1
 2 ...***..*........*........**.***..* 2
 3 ...***..*..*.....*........**.***..* 3
 4 ...***..**.*.....*........**.***..* 4
 5 ...***..**.*.....*........**.***..* 5
 6 ...***..**.*.....*........**.***..* 6
 7 ...***..****.....***......**.***.** 7
 8 ...***..****...*.***......**.***.** 8
 9 ...***..****.*.*.***......**.***.** 9
 A ...***..****.*.*.***......**.***.** A
 B ...***..****.*.*****..*...**.****** B
 C ...***..****.*.*****..*.*.********* C
 D ...***..****.*.*****..***.********* D
 E ...***..******.*****.************** E
 F ...***..************.************** F

   0123456789ABCDEF0123456789ABCDEF012
```

```
SOURCE FILE: DISKOS
0024:             1 CH       EQU  $24
0025:             2 CV       EQU  $25
00FA:             3 REGIS    EQU  $FA
00FC:             4 V.AL     EQU  $FC
00FD:             5 V.AH     EQU  $FD
00FE:             6 V.BL     EQU  $FE
00FF:             7 V.BH     EQU  $FF
8400:             8 VTOCBUF  EQU  $8400
8900:             9 CTLGBUF  EQU  $8900
AE42:            10 VOL3DIG  EQU  $AE42
B7B5:            11 RWTSCAL  EQU  $B7B5
FC22:            12 VTAB     EQU  $FC22
FDBE:            13 CROUT    EQU  $FDBE
FDDA:            14 PRBYTE   EQU  $FDDA
FDED:            15 COUT     EQU  $FDED
----- NEXT OBJECT FILE NAME IS DISKOS.OBJ0
8500:            16          ORG  $8500
8500:4C 51 85    17 DISKOS   JMP  LDVTOC
8503:4C 72 85    18          JMP  LDCTLG
8506:4C 6D 85    19          JMP  SVVTOC
8509:4C 9A 85    20          JMP  SVCTLG
850C:4C 9F 85    21          JMP  TSGRAPH
850F:00 00       22 FREE     DFB  0,0
8511:A9 00       23 CLRFREE  LDA  #0
8513:8D 0F 85    24          STA  FREE
8516:8D 10 85    25          STA  FREE+1
8519:60          26          RTS
851A:EE 0F 85    27 INCFREE  INC  FREE
851D:D0 03       28          BNE  INCFRE1
851F:EE 10 85    29          INC  FREE+1
8522:60          30 INCFRE1  RTS
8523:A9 85       31 IBCNTRL  LDA  #<IBTYPE
8525:A0 2D       32          LDY  #>IBTYPE
8527:20 B5 B7    33          JSR  RWTSCAL
852A:4C 51 A8    34          JMP  $A851
852D:01          35 IBTYPE   DFB  $01
852E:60          36 IBSLOT   DFB  $60
852F:01          37 IBDRVN   DFB  $01
8530:00          38 IBVOL    DFB  $00
8531:00          39 IBTRK    DFB  $00
8532:00          40 IBSECT   DFB  $00
8533:3E          41 IBDCTP   DFB  >DEVTPC
8534:85          42          DFB  <DEVTPC
8535:00 00       43 IBBUFF   DFB  $00,$00
8537:00 00       44          DFB  $00,$00
8539:00          45 IBCMD    DFB  $00
853A:00          46 IBSTAT   DFB  $00
853B:00          47 IBSMOD   DFB  $00
853C:60          48 IOBPSN   DFB  $60
853D:01          49 IOBPDN   DFB  $01
853E:00          50 DEVTPC   DFB  $00
853F:01          51 PPTC     DFB  $01
8540:EF D8       52 MONTC    DFB  $EF,$D8
8542:4C C7 86    53          JMP  L.1800
8545:00          54 V.CE     DFB  0
8546:00          55 V.CH     DFB  0
8547:00          56 V.CL     DFB  0
8548:00          57 V.CNL    DFB  0
8549:00          58 V.CNH    DFB  0
854A:00          59 V.F      DFB  0
854B:00          60 V.FZ     DFB  0
854C:00          61 V.P      DFB  0
854D:00          62 V.E      DFB  0
854E:4C 56 87    63          JMP  DSPLCAT
8551:A9 01       64 LDVTOC   LDA  #$01
8553:8D 39 85    65 LDVTOC1  STA  IBCMD
8556:A9 00       66          LDA  #$00
8558:8D 32 85    67          STA  IBSECT
855B:A9 11       68          LDA  #$11
855D:8D 31 85    69          STA  IBTRK
8560:A9 00       70          LDA  #>VTOCBUF
8562:8D 35 85    71          STA  IBBUFF
8565:A9 84       72          LDA  #<VTOCBUF
8567:8D 36 85    73          STA  IBBUFF+1
856A:4C 23 85    74          JMP  IBCNTRL
856D:A9 02       75 SVVTOC   LDA  #$02
856F:4C 53 85    76          JMP  LDVTOC1
8572:A9 01       77 LDCTLG   LDA  #$01
8574:8D 39 85    78 LDCTLG1  STA  IBCMD
8577:A9 11       79          LDA  #$11
8579:8D 31 85    80          STA  IBTRK
857C:A9 0F       81          LDA  #$0F
857E:8D 32 85    82          STA  IBSECT
8581:A9 00       83          LDA  #>CTLGBUF
8583:8D 35 85    84          STA  IBBUFF
8586:A9 89       85          LDA  #<CTLGBUF
8588:8D 36 85    86          STA  IBBUFF+1
858B:20 23 85    87 LDCTLG2  JSR  IBCNTRL
858E:CE 32 85    88          DEC  IBSECT
8591:F0 06       89          BEQ  LDCTLGX
8593:EE 36 85    90          INC  IBBUFF+1
8596:4C 8B 85    91          JMP  LDCTLG2
8599:60          92 LDCTLGX  RTS
859A:A9 02       93 SVCTLG   LDA  #$02
859C:4C 74 85    94          JMP  LDCTLG1
859F:A9 38       95 TSGRAPH  LDA  #$38
85A1:8D FE 87    96          STA  XSAV
85A4:A9 03       97          LDA  #$03
85A6:8D FF 87    98          STA  CHSAV
85A9:20 11 85    99          JSR  CLRFREE
85AC:A9 15      100 TSGR1    LDA  #21
85AE:8D 00 88   101          STA  CVSAV
85B1:AE FE 87   102          LDX  XSAV
85B4:BD 00 84   103          LDA  VTOCBUF,X
85B7:8D 01 88   104          STA  BYTE
85BA:20 3A 86   105 TSGR2    JSR  TSGR10
85BD:B0 23      106          BCS  TSGR4
85BF:AD 05 84   107          LDA  VTOCBUF+5
85C2:D0 0C      108          BNE  TSGR2.5
85C4:AD FF 87   109          LDA  CHSAV
85C7:C9 06      110          CMP  #6
85C9:B0 05      111          BCS  TSGR2.5
85CB:A9 A3      112          LDA  #$A3
85CD:4C D2 85   113          JMP  TSGR3
85D0:A9 AA      114 TSGR2.5  LDA  #$AA
85D2:20 ED FD   115 TSGR3    JSR  COUT
85D5:CE 00 88   116          DEC  CVSAV
85D8:AD 00 88   117          LDA  CVSAV
85DB:C9 0D      118          CMP  #13
85DD:F0 0B      119          BEQ  TSGR4.1
85DF:4C BA 85   120          JMP  TSGR2
85E2:20 1A 85   121 TSGR4    JSR  INCFREE
85E5:A9 AE      122          LDA  #$AE
85E7:4C D2 85   123          JMP  TSGR3
85EA:EE FE 87   124 TSGR4.1  INC  XSAV
85ED:AE FE 87   125          LDX  XSAV
85F0:BD 00 84   126          LDA  VTOCBUF,X
85F3:8D 01 88   127          STA  BYTE
85F6:20 3A 86   128 TSGR5    JSR  TSGR10
```

```
85F9:B0 23     129          BCS  TSGR7
85FB:AD 05 84  130          LDA  VTOCBUF+5
85FE:D0 0C     131          BNE  TSGR5.5
8600:AD FF 87  132          LDA  CHSAV
8603:C9 06     133          CMP  #6
8605:B0 05     134          BCS  TSGR5.5
8607:A9 A3     135          LDA  #$A3
8609:4C 0E 86  136          JMP  TSGR6
860C:A9 AA     137 TSGR5.5  LDA  #$AA
860E:20 ED FD  138 TSGR6    JSR  COUT
8611:CE 00 88  139          DEC  CVSAV
8614:AD 00 88  140          LDA  CVSAV
8617:C9 05     141          CMP  #5
8619:F0 0B     142          BEQ  TSGR8
861B:4C F6 85  143          JMP  TSGR5
861E:20 1A 85  144 TSGR7    JSR  INCFREE
8621:A9 AE     145          LDA  #$AE
8623:4C 0E 86  146          JMP  TSGR6
8626:EE FF 87  147 TSGR8    INC  CHSAV
8629:AD FE 87  148          LDA  XSAV
862C:18        149          CLC
862D:69 03     150          ADC  #3
862F:8D FE 87  151          STA  XSAV
8632:C9 C4     152          CMP  #$C4
8634:B0 03     153          BCS  TSGR9
8636:4C AC 85  154          JMP  TSGR1
8639:60        155 TSGR9    RTS
863A:AD 00 88  156 TSGR10   LDA  CVSAV
863D:85 25     157          STA  CV
863F:AD FF 87  158          LDA  CHSAV
8642:85 24     159          STA  CH
8644:20 22 FC  160          JSR  VTAB
8647:0E 01 88  161          ASL  BYTE
864A:60        162          RTS
864B:A9 00     163 L.1700   LDA  #0
864D:8D 46 85  164          STA  V.CH
8650:8D 47 85  165          STA  V.CL
8653:AD 45 85  166          LDA  V.CE
8656:C9 07     167 L.1700A  CMP  #7
8658:90 08     168          BCC  L.1700B
865A:E9 07     169          SBC  #7
865C:EE 46 85  170          INC  V.CH
865F:4C 56 86  171          JMP  L.1700A
8662:8D 47 85  172 L.1700B  STA  V.CL
8665:AD 47 85  173 L.1705   LDA  V.CL
8668:8D 48 85  174          STA  V.CNL
866B:A9 00     175          LDA  #0
866D:8D 49 85  176          STA  V.CNH
8670:A2 05     177          LDX  #5
8672:0E 48 85  178 L.1705A  ASL  V.CNL
8675:2E 49 85  179          ROL  V.CNH
8678:CA        180          DEX
8679:D0 F7     181          BNE  L.1705A
867B:A2 03     182          LDX  #3
867D:18        183 L.1705B  CLC
867E:AD 48 85  184          LDA  V.CNL
8681:6D 47 85  185          ADC  V.CL
8684:8D 48 85  186          STA  V.CNL
8687:AD 49 85  187          LDA  V.CNH
868A:69 00     188          ADC  #0
868C:8D 49 85  189          STA  V.CNH
868F:CA        190          DEX
8690:D0 EB     191          BNE  L.1705B
8692:18        192          CLC
8693:AD 48 85  193          LDA  V.CNL
8696:69 0B     194          ADC  #11
```

```
8698:8D 48 85  195          STA  V.CNL
869B:AD 49 85  196          LDA  V.CNH
869E:6D 46 85  197          ADC  V.CH
86A1:8D 49 85  198          STA  V.CNH
86A4:18        199          CLC
86A5:AD 48 85  200          LDA  V.CNL
86A8:69 00     201          ADC  #>CTLGBUF
86AA:8D 48 85  202          STA  V.CNL
86AD:AD 49 85  203          LDA  V.CNH
86B0:69 89     204          ADC  #<CTLGBUF
86B2:8D 49 85  205          STA  V.CNH
86B5:AD 48 85  206 L.1710   LDA  V.CNL
86B8:85 FA     207          STA  REGIS
86BA:AD 49 85  208          LDA  V.CNH
86BD:85 FB     209          STA  REGIS+1
86BF:A0 02     210          LDY  #2
86C1:B1 FA     211          LDA  (REGIS),Y
86C3:8D 4C 85  212          STA  V.P
86C6:60        213          RTS
86C7:A9 01     214 L.1800   LDA  #1
86C9:8D 45 85  215          STA  V.CE
86CC:A9 00     216          LDA  #0
86CE:8D 4A 85  217          STA  V.F
86D1:8D 4B 85  218          STA  V.FZ
86D4:20 4B 86  219 L.1805   JSR  L.1700
86D7:AD 48 85  220          LDA  V.CNL
86DA:85 FC     221          STA  V.AL
86DC:AD 49 85  222          LDA  V.CNH
86DF:85 FD     223          STA  V.AH
86E1:EE 45 85  224          INC  V.CE
86E4:20 4B 86  225          JSR  L.1700
86E7:AD 48 85  226          LDA  V.CNL
86EA:85 FE     227          STA  V.BL
86EC:AD 49 85  228          LDA  V.CNH
86EF:85 FF     229          STA  V.BH
86F1:AD 4A 85  230          LDA  V.F
86F4:D0 0D     231          BNE  L.1810
86F6:A0 00     232          LDY  #0
86F8:B1 FC     233          LDA  (V.AL),Y
86FA:F0 07     234          BEQ  L.1810
86FC:C9 24     235          CMP  #36
86FE:B0 03     236          BCS  L.1810
8700:EE 4B 85  237          INC  V.FZ
8703:A0 00     238 L.1810   LDY  #0
8705:B1 FC     239          LDA  (V.AL),Y
8707:C9 FF     240          CMP  #$FF
8709:F0 2A     241          BEQ  L.1870
870B:A0 00     242 L.1815   LDY  #0
870D:B1 FE     243          LDA  (V.BL),Y
870F:F0 1B     244          BEQ  L.1865
8711:A0 03     245 L.1835   LDY  #3
8713:B1 FC     246 L.1840   LDA  (V.AL),Y
8715:D1 FE     247          CMP  (V.BL),Y
8717:F0 04     248          BEQ  L.1850
8719:90 07     249          BCC  L.1860
871B:B0 18     250          BCS  L.1870
871D:C8        251 L.1850   INY
871E:C0 21     252          CPY  #33
8720:90 F1     253          BCC  L.1840
8722:AD 45 85  254 L.1860   LDA  V.CE
8725:AD 45 85  255          LDA  V.CE
8728:C9 69     256          CMP  #105
872A:90 A8     257          BCC  L.1805
872C:AD 4A 85  258 L.1865   LDA  V.F
872F:D0 01     259          BNE  L.1866
8731:60        260          RTS
```

```
8732:4C C7 86  261 L.1866   JMP  L.1800
8735:A0 00     262 L.1870   LDY  #0
8737:B1 FE     263          LDA  (V.BL),Y
8739:F0 E7     264          BEQ  L.1860
873B:C9 FF     265          CMP  #255
873D:F0 E3     266          BEQ  L.1860
873F:A0 22     267 L.1873   LDY  #34
8741:B1 FC     268 L.1873A  LDA  (V.AL),Y
8743:48        269          PHA
8744:B1 FE     270          LDA  (V.BL),Y
8746:91 FC     271          STA  (V.AL),Y
8748:68        272          PLA
8749:91 FE     273          STA  (V.BL),Y
874B:88        274          DEY
874C:10 F3     275          BPL  L.1873A
874E:A9 01     276 L.1875   LDA  #1
8750:8D 4A 85  277          STA  V.F
8753:4C 22 87  278          JMP  L.1860
8756:A9 01     279 DSPLCAT  LDA  #1
8758:8D 4D 85  280          STA  V.E
875B:20 4B 86  281 L.315    JSR  L.1700
875E:AD 48 85  282          LDA  V.CNL
8761:85 FC     283          STA  V.AL
8763:AD 49 85  284          LDA  V.CNH
8766:85 FD     285          STA  V.AH
8768:A0 00     286          LDY  #0
876A:B1 FC     287          LDA  (V.AL),Y
876C:D0 01     288          BNE  L.315A
876E:60        289          RTS
876F:C9 FF     290 L.315A   CMP  #$FF
8771:D0 01     291          BNE  L.320
8773:60        292          RTS
8774:AD 4C 85  293 L.320    LDA  V.P
8777:10 08     294          BPL  L.325
8779:A9 AA     295          LDA  #$AA
877B:20 ED FD  296          JSR  COUT
877E:4C 86 87  297          JMP  L.330
8781:A9 A0     298 L.325    LDA  #$A0
8783:20 ED FD  299          JSR  COUT
8786:AD 4C 85  300 L.330    LDA  V.P
8789:29 7F     301          AND  #$7F
878B:D0 05     302          BNE  L.335
878D:A9 D4     303          LDA  #$D4
878F:4C B8 87  304          JMP  L.345C
8792:C9 01     305 L.335    CMP  #1
8794:D0 05     306          BNE  L.340
8796:A9 C9     307          LDA  #$C9
8798:4C B8 87  308          JMP  L.345C
879B:C9 02     309 L.340    CMP  #2
879D:D0 05     310          BNE  L.345
879F:A9 C1     311          LDA  #$C1
87A1:4C B8 87  312          JMP  L.345C
87A4:C9 04     313 L.345    CMP  #4
87A6:D0 05     314          BNE  L.345A
87A8:A9 C2     315          LDA  #$C2
87AA:4C B8 87  316          JMP  L.345C
87AD:C9 10     317 L.345A   CMP  #16
87AF:D0 05     318          BNE  L.345B
87B1:A9 D2     319          LDA  #$D2
87B3:4C B8 87  320          JMP  L.345C
87B6:A9 A0     321 L.345B   LDA  #$A0
87B8:20 ED FD  322 L.345C   JSR  COUT
87BB:A9 A0     323          LDA  #$A0
87BD:20 ED FD  324          JSR  COUT
87C0:A0 21     325 L.350    LDY  #33
87C2:B1 FC     326          LDA  (V.AL),Y
```

```
87C4:85 44     327          STA  $44
87C6:C8        328          INY
87C7:B1 FC     329          LDA  (V.AL),Y
87C9:85 45     330          STA  $45
87CB:20 42 AE  331          JSR  VOL3DIG
87CE:A9 A0     332          LDA  #$A0
87D0:20 ED FD  333          JSR  COUT
87D3:AD 4D 85  334          LDA  V.E
87D6:20 ED FD  335          JSR  COUT
87D9:A9 A0     336          LDA  #$A0
87DB:20 ED FD  337          JSR  COUT
87DE:A0 03     338 L.355    LDY  #3
87E0:B1 FC     339 L.355A   LDA  (V.AL),Y
87E2:20 ED FD  340          JSR  COUT
87E5:C8        341          INY
87E6:C0 21     342          CPY  #33
87E8:90 F6     343          BCC  L.355A
87EA:20 8E FD  344          JSR  CROUT
87ED:EE 4D 85  345          INC  V.E
87F0:EE 45 85  346          INC  V.CE
87F3:AD 4D 85  347          LDA  V.E
87F6:C9 11     348          CMP  #17
87F8:90 01     349          BCC  L.355B
87FA:60        350          RTS
87FB:4C 5B 87  351 L.355B   JMP  L.315
87FE:00        352 XSAV     DFB  0
87FF:00        353 CHSAV    DFB  0
8800:00        354 CVSAV    DFB  0
8801:00        355 BYTE     DFB  0

*** SUCCESSFUL ASSEMBLY: NO ERRORS
```

```
-------- DOS 3.3  DISK-HANDLING --------
VOL DOS LOAD RUN  NXTPG LOCK UNLOCK FULL
RENAME BACKUP COPY DELETE GRAPH NEW EXIT

VOL 001 PG 2 OF 2 SECT-USED 558 LEFT 002
----------------------------------------

*B 005 A DISKOS.OBJO
*A 010 B EXEC DEMO
*B 020 C FID
*B 050 D FPBASIC
*B 002 E HDC
*A 006 F HELLO (DISK-ORIGINAL)
*B 050 G INTBASIC
*A 028 H LITTLE BRICK OUT
*A 003 I MAKE TEXT
*B 009 J MASTER CREATE
*B 027 K MUFFIN
*A 051 L PHONE LIST
*A 010 M RANDOM
*A 013 N RENUMBER
*A 039 O RENUMBER INSTRUCTIONS
*A 003 P RETRIEVE TEXT
```

```
-------- DOS 3.3  DISK-HANDLING --------
VOL DOS LOAD RUN  NXTPG LOCK UNLOCK FULL
RENAME BACKUP COPY DELETE GRAPH NEW EXIT

VOL 001 PG 1 OF 2 SECT-USED 558 LEFT 002
----------------------------------------

*A 002 A HELLO
*I 018 B ANIMALS
*T 003 C APPLE PROMS
*I 006 D APPLESOFT
*I 026 E APPLEVISION
*I 017 F BIORHYTHM
*B 010 G BOOT13
*A 006 H BRIAN'S THEME
*B 003 I CHAIN
*I 009 J COLOR DEMO
*A 009 K COLOR DEMOSOFT
*I 009 L COPY
*B 003 M COPY.OBJO
*A 009 N COPYA
*A 019 O DISK-HANDLING
*T 019 P DISKOS
```



VC-20/HC - Der Zeichen-Generator

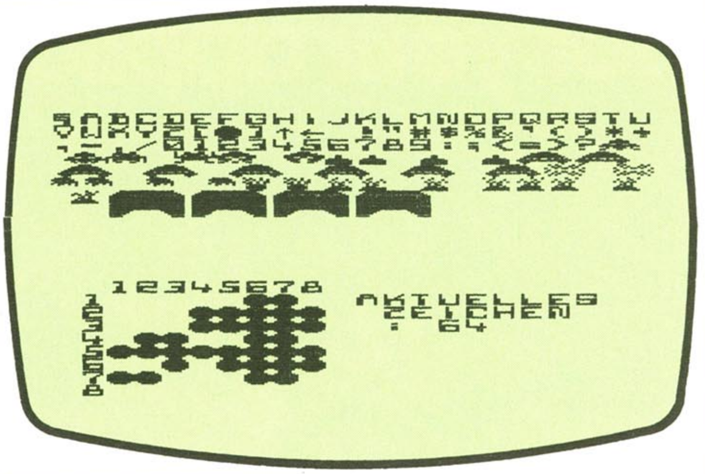
**Eine der interessantesten Fähigkeiten des VC-20 ist sein programmierbarer Zeichensatz.
Homecomputer zeigt, wie man diese schöne Sache komfortabel und erfolgreich in Programmen anwenden kann.**

Wenn man sich für eigene Programme selbst eigene Zeichen definieren möchte, stößt man zunächst auf einige Probleme.
Es gibt nämlich nur wenige Bereiche, in denen sich der selbstdefinierte Zeichensatz ablegen läßt, um vom Betriebssystem erkannt zu werden.
Sämtliche Bereiche sind entweder normalerweise vom Programm oder vom Bildschirmspeicher belegt oder sind bei einer Erweiterung verloren gegangen.
Damit möglichst alle Anwender die Homecomputer-Programme ohne sie umschreiben zu müssen benutzen können, mußte deshalb eine Art Standard gefunden werden. Programme, die auf eigene Zeichensätze zugreifen, werden deshalb in Homecomputer möglichst nach den hier aufgestellten Regeln erstellt.

Das bedeutet zum Beispiel, daß der Zeichensatz grundsätzlich wie ein Basic-Programm (oder als Teil eines solchen) geladen wird, daß er sich immer zwischen Startprogramm und Hauptprogramm befindet.

POKE 44,28
POKE 7168,0

Vor Eintasten des Programms wird mit POKE44,28 : POKE7168,0 der Basic-Start-Zeiger auf den neuen Hauptprogrammanfang eingestellt.
Das Gleiche wird übrigens jedesmal wiederholt, wenn der auf Kassette oder Diskette gerettete Zeichensatz wieder geladen wird.

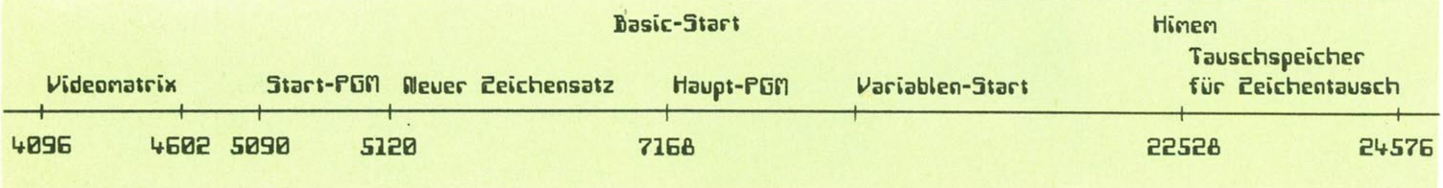


REMarks zum Programm

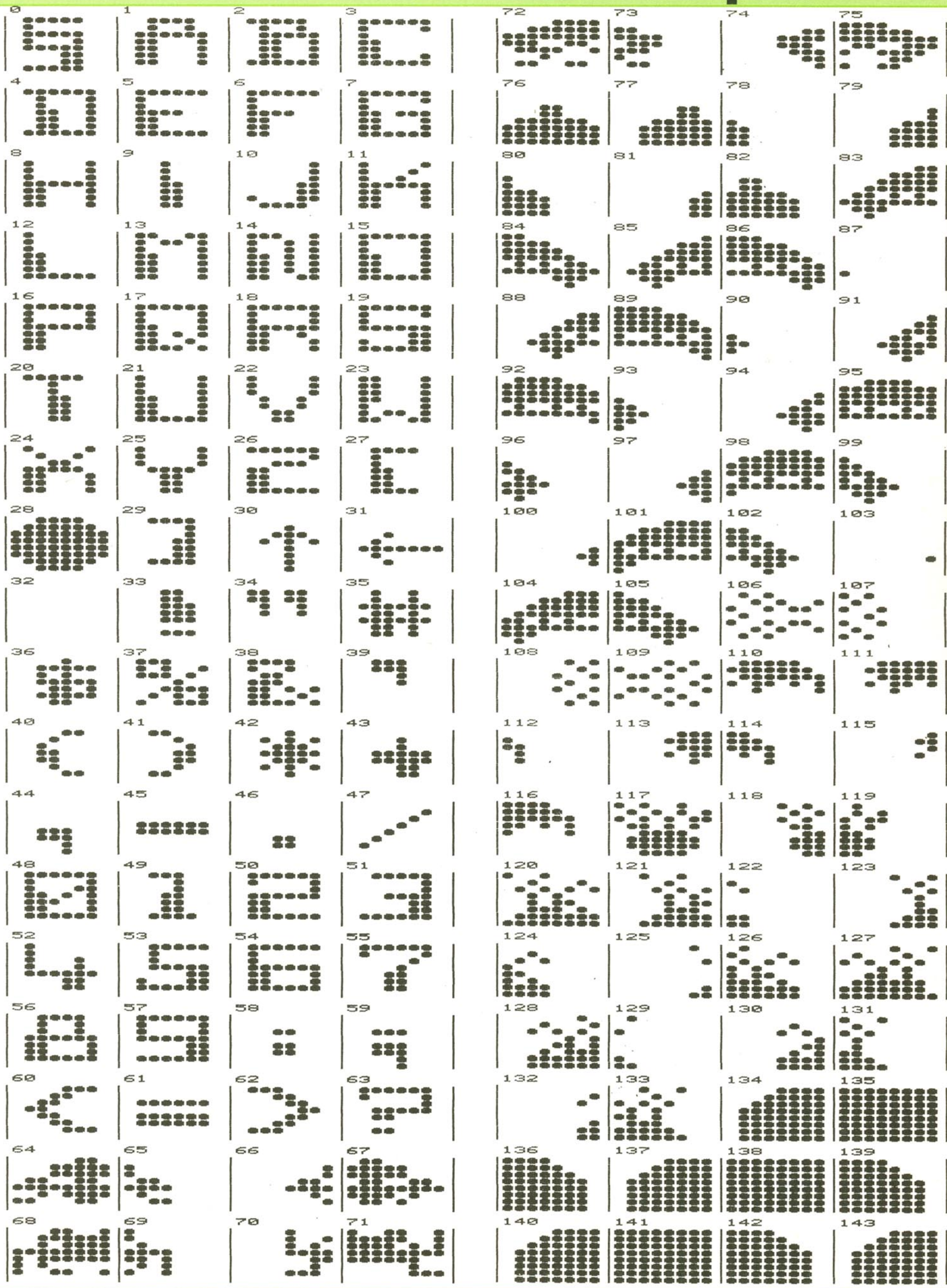
(11) HIMEM setzen.
(12-14) Startprogramm vor neuen Zeichensatz poken.
(20) Variable:
ZS=Zeichensatz (neu)
OZ=Originalzeichensatz
KZ=Kleinschriftsatz
TS=Tauschspeicher
VM=Videomatrix
FM=Farbmatrix
(22-38) Kompletten Originalsatz oder davon nur Zeichen 0-64 in neuen Satz übernehmen.
(40) Zeiger auf neuen Zeichensatz
(50) Display Zeichensatz
(60-300) Kommandos interpretieren.
(1000-1060) Display vergrößertes Zeichen.
(2000-2070) Save Zeichensatz auf Disk oder Kassettenrecorder.
(3000-3040) Load Zeichensatz von Disk oder Kassettenrecorder.
(4000-4040) Neues Zeichen durch Eingabe von Ziffern.
(4500-4540) Zwei Zeichen gleichsetzen.
(4600-7000) Display-Routinen.
(8000-8010) Tauschspeicher Zeichen in neuen Zeichensatz übernehmen.
(9000-9070) Suchen der Programmspeicherstellen, in die der Zeichensatzname gepoked wird. Namen eingeben und speichern.
(9500-9610) Drucke gesamten Bildschirm mit definiertem Zeichensatz auf Seikosha GP-100VC
(10000-10100) Neuen Zeichensatz auf Drucker listen.

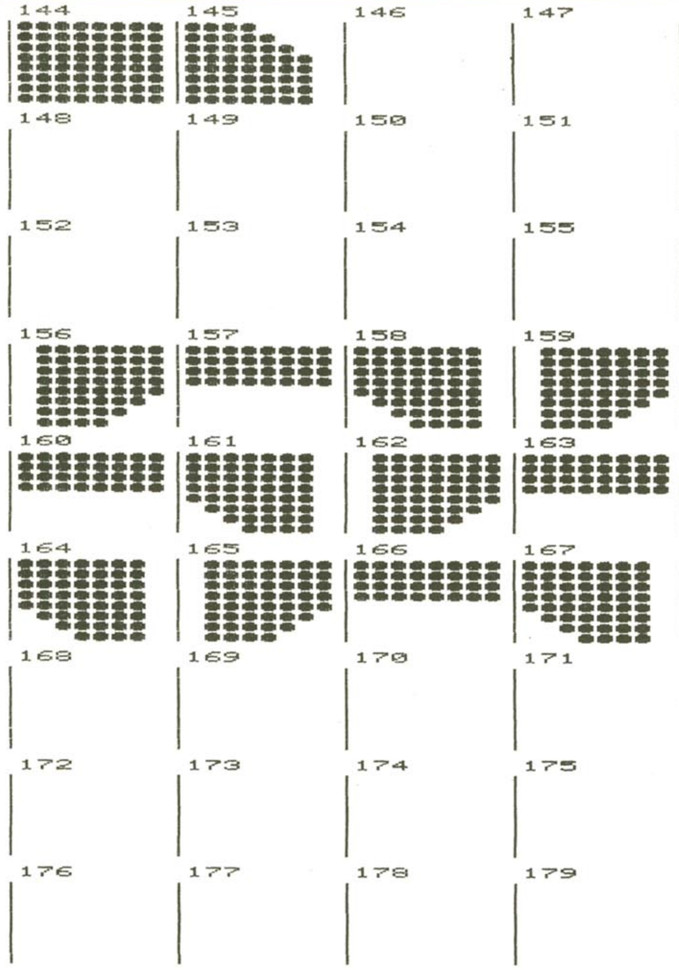
Kommandos

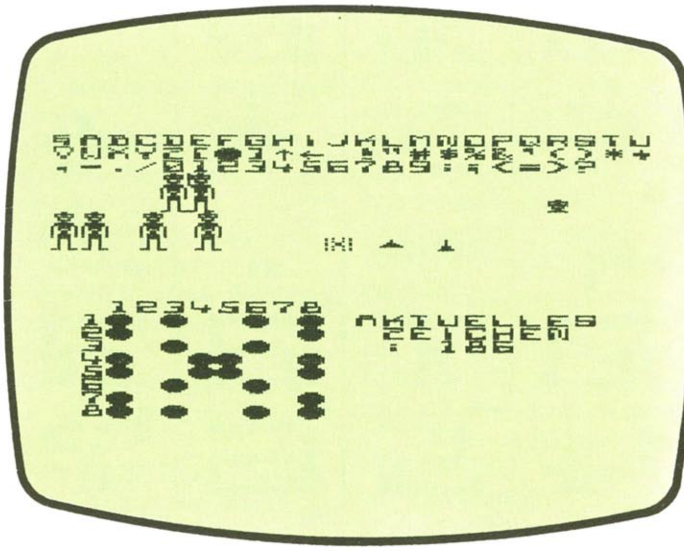
<I> Cursor nach oben
<J> Cursor nach links
<K> Cursor nach rechts
<M> Cursor nach unten
<Q> Punkt setzen
<W> Punkt löschen
<=> Dupliziere Zeichen
<Zahl 0 - 255> Aktuelles Zeichen
<SHIFT> <L> Lade Zeichensatz von Disk bzw. Kassette
<SHIFT> <S> Save Zeichensatz auf Disk bzw. Kassette
<X> Exit (Verlasse) Zeichengenerator mit definiertem Zeichensatz.
<C> (Clear) Loesche Zeichen 64 -255
<SHIFT> <X> Exit mit Original-Zeichensatz
<SHIFT> <T> Tausche Zei-



Speicherplatzbelegung (VC-20 + 16K)







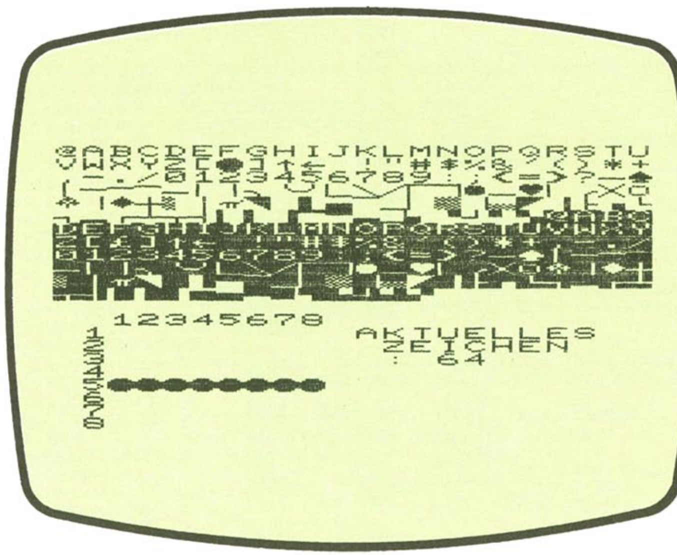
chensatz mit Tauschspeicher
<T> Tausche aktuelles Zeichen mit entsprechendem Tauschspeicherzeichen
<SHIFT> <V> Verschiebe definiertes Zeichen in Tauschspeicher
<SHIFT> <G> Originalsatz Groß/Grafik wird in Buffer für neuen Zeichensatz gepoked
<SHIFT> <K> Kleinschrift-satz wird in Buffer für neuen Zeichensatz gepoked
<SHIFT> <N> Name des Zeichensatzes für Load und Save eingeben
<SHIFT> <P> Print Zeichen in Groß-Grafik auf Drucker
<SHIFT> <#> Drucke kompletten Bildschirm mit definierten Zeichen auf Seikosha GP-100VC

Starship VC-20

```
10 PRINT"□"
11 RESTORE:POKE56,86:CLR
12 DATA0,254,19,1,0,151,52,51,44,49,58,1
51,52,52,44,50,56,58
13 DATA151,55,49,54,56,44,48,58,138,0,0,
0
14 FORI=5090TO5119:READJ:POKEI,J:NEXT
20 ZS=5120:OZ=32768:KZ=34816:TS=22528:VM
=4096:FM=37888:DIMM(7,7),P(3)
22 PRINT"ORIGINAL-ZEICHENSATZ:    K = KOM
PLETT              B = NUR BUCHSTABEN   ";
23 PRINT"   N = NICHTS AENDERN       ";
25 GETA$:IFA$=""THEN25
26 IFA$="K"THENM=2047:GOTO30
27 IFA$="B"THENM=511:GOTO30
28 IFA$="N"THEN40
29 GOTO10
30 FORI=0TOM:POKEI+ZS,PEEK(OZ+I):NEXT
36 POKEZS+224,60:POKEZS+225,126:POKEZS+2
26,255:POKEZS+227,255
37 POKEZS+231,60:POKEZS+230,126:POKEZS+2
29,255:POKEZS+228,255
38 POKE5117,0:POKE5118,0:POKE5119,0
40 POKE36869,205:POKE650,128
50 FORI=0TO255:POKEVM+I,I:POKEFM+I,6:NEX
T
60 CZ=32:ZN=64:GOSUB5000
70 F=0:GOSUB1000
100 REM
101 IFCZ=32THENCZ=28:GOTO103
102 IFCZ=28THENCZ=32
103 POKEVM+(CY+14)*22+CX+2,CZ:POKEFM+(CY
+14)*22+CX+2,6
104 GETA$:IFA$=""THEN100
105 GOSUB7000:F=0
110 IFA$="I"THENF=1:CY=CY-1:IFCY<0THENCY
=0
120 IFA$="J"THENF=1:CX=CX-1:IFCX<0THENCX
=0
130 IFA$="K"THENF=1:CX=CX+1:IFCX>7THENCX
=7
140 IFA$="M"THENF=1:CY=CY+1:IFCY>7THENCY
=7
145 GOSUB7000
150 IFA$="Q"THENF=1:M(CX,CY)=28:POKEZS+Z
N*8+CY,PEEK(ZS+ZN*8+CY)OR2↑(7-CX):GOSUB7
000
160 IFA$="W"THENF=1:M(CX,CY)=32:POKEZS+Z
N*8+CY,PEEK(ZS+ZN*8+CY)AND255-2↑(7-CX):G
OSUB7000
170 IFA$="♠"THENPRINT"□":POKE36869,192:P
OKE650,0:END
172 IFA$="X"THENEND
180 IFF=1THEN100
200 IFA$="*"THEN2000
210 IFA$="L"THEN3000
212 IFA$="-"THENFORI=5632TO7167:POKEI,0:
NEXT:F=0:GOSUB1000:GOTO100
215 IFA$="="THENGOSUB4500
220 IFASC(A$)>47ANDASC(A$)<58THENGOSUB40
00:GOTO100
230 IFA$="X"THENFORI=0TO2047:POKETS+I,PE
EK(ZS+I):NEXT:GOTO100
240 IFA$="|"THENFORI=0TO2047:POKEZS+I,PE
EK(OZ+I):NEXT:GOTO100
250 IFA$="/"THENFORI=0TO2047:POKEZS+I,PE
EK(KZ+I):NEXT:GOTO100
255 IFA$="#"THENGOSUB9500:GOTO100
260 IFA$="T"THENGOSUB8000:GOTO100
270 IFA$="|"THENF=1:ZZ=ZN:FORZN=0TO255:G
OSUB8000:NEXT:ZN=ZZ:GOTO100
280 IFA$="/"THENGOSUB9000:GOTO40
290 IFA$="┐"THENGOSUB10000:GOTO40
300 GOTO100
1000 PRINT"▓▓▓▓▓▓▓▓▓▓▓▓▓▓   12345678□"
1005 IFF=1THENJ=CY:FORI=0TOCY:PRINT"▣";:
NEXT:GOTO1020
1010 PRINT:FORJ=0TO7
1020 P=PEEK(ZS+ZN*8+J):PRINTJ+1;"■";
1030 FORI=7TO0STEP-1:IFP>=2↑ITHENPRINT"£
";:P=P-2↑I:M(7-I,J)=28:GOTO1050
1040 PRINT" ";:M(7-I,J)=32
1050 NEXT:PRINT:IFF=0THENNEXT
1060 RETURN
2000 PRINT"□":POKE36869,192
2001 PRINT"<D> = DISK      "
2002 INPUT"<K> = KASSETTE";FR$
2005 FORI=0TO3:P(I)=PEEK(I+43):NEXT
2015 IFFR$="K"THEN2049
2016 IFFR$="D"THEN2019
2017 GOTO2030
2019 POKE43,227:POKE44,19:POKE45,0:POKE4
6,28
2020 SAVE"@:ROBOT.ZCHN          ",8
2030 POKE43,P(0):POKE44,P(1):POKE45,P(2)
:POKE46,P(3)
2040 RUN
2049 POKE43,227:POKE44,19:POKE45,0:POKE4
6,28
2050 SAVE"ROBOT.ZCHN            "
2060 POKE43,P(0):POKE44,P(1):POKE45,P(2)
:POKE46,P(3)
2070 RUN
3000 PRINT"□":POKE36869,192
3001 PRINT"<D> = DISK      "
3002 INPUT"<K> = KASSETTE";FR$
3005 FORI=0TO3:P(I)=PEEK(I+43):NEXT
3010 POKE43,227:POKE44,19
3015 IFFR$="K"THEN3040
3016 IFFR$<>"D"THEN2030
3020 LOAD"ROBOT.ZCHN          ",8
3030 END
3040 LOAD"ROBOT.ZCHN          "
4000 ZN=VAL(A$)
4005 GOSUB5000
4010 GETA$:IFA$=""THEN4010
4020 IFASC(A$)=13THENF=0:GOSUB1000:CX=0:
CY=0:RETURN
4030 ZN=ZN*10+VAL(A$):IFZN>255THENZN=0:G
OTO4005
4040 GOTO4005
4500 WZ=0
4505 GOSUB6000
4510 GETA$:IFA$=""THEN4510
4520 IFASC(A$)=13THENGOSUB4600:F=0:GOSUB
1000:CX=0:CY=0:RETURN
```

```
4530 WZ=WZ*10+VAL(A$):IFWZ>255THEN4500
4540 GOTO4505
4600 FORI=0TO7:POKEZS+I+ZN*8,PEEK(ZS+I+W
Z*8):NEXT
4610 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";
4620 PRINT"■■■■■■■■■■■■■■■               ■■■■■■■■
■■■■■"
4630 RETURN
5000 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";
5005 PRINT"■■■■■■■■■■■AKTUELLES■■■■■■■■
■■■■■ZEICHEN■■■■■■■■■■■■■■■■■■:";ZN;"■■    "
5010 RETURN
6000 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";
6005 PRINT"■■■■■■■■■■■■■■   WIRD   ■■■■■■■■■■■
■■■■■:";WZ;"■■    "
6010 RETURN
7000 POKEVM+(CY+14)*22+CX+2,M(CX,CY):RET
URN
8000 FORI=0TO7:X=PEEK(ZS+ZN*8+I):POKEZS+
ZN*8+I,PEEK(TS+ZN*8+I):POKETS+ZN*8+I,X:N
EXT
8010 IFF=0THENGOSUB1000
8020 RETURN
9000 A=PEEK(43)+PEEK(44)*256
9010 NZ=PEEK(A)+PEEK(A+1)*256:ZN=PEEK(A+
2)+PEEK(A+3)*256
9015 IFZN<2020THEN9030
9020 IFZN=2020THENPS=A+7
9021 IFZN=2050THENKS=A+5
9025 IFZN=3020THENPL=A+5
9026 IFZN=3040THENKL=A+5:GOTO9040
9030 A=NZ:GOTO9010
9040 PRINT"▒":POKE36869,192:FORI=1TO16:X
=PEEK(PS+I):PRINTCHR$(X);:NEXT:PRINT:PRI
NT
9045 INPUT"NAME";NA$:IFLEN(NA$)>16THENPR
INTLEN(NA$)-16;" ZEICHEN ZU LANG!":GOTO9
040
9050 IFLEN(NA$)<16THENNA$=NA$+" ":GOTO90
50
9060 FORI=1TO16:X=ASC(MID$(NA$,I)):POKEP
S+I,X:POKEPL+I,X:POKEKS+I,X:POKEKL+I,X:N
EXT
9070 POKE36869,205:RETURN
9500 PB=22016
9502 OPEN4,4
9505 PRINT#4,CHR$(14);NA$
9510 PRINT#4,CHR$(8)
9515 DL=0
9520 DZ=0:FORI=0TO352:POKEPB+I,128:NEXT
9530 ZX=INT(DL/8):ZR=DL-ZX*8:Y=2↑DZ:BZ=P
B
9540 FORZY=0TO21:Z=PEEK(VM+ZX*22+ZY):ZT=
PEEK(ZS+Z*8+ZR)
9550 FORI=7TO0STEP-1:IFZT<2↑ITHEN9560
9555 ZT=ZT-2↑I:POKEBZ,PEEK(BZ)+Y:POKEBZ+
1,PEEK(BZ+1)+Y
9560 BZ=BZ+2:NEXT:NEXT
9570 DL=DL+1
9580 IFDL=182THENGOSUB9600:PRINT#4,CHR$(
15):CLOSE4:RETURN
9585 DZ=DZ+1:IFDZ=7THENGOSUB9600:GOTO952
0
9590 GOTO9530
9600 PRINTCHR$(8);:FORI=0TO351:PRINT#4,C
HR$(PEEK(PB+I));:NEXT:PRINT#4,CHR$(8)
9610 RETURN
10000 PG=4:ZP=ZS:I$(0)="00":I$(1)="18":I
$(2)="36":I$(3)="54":B$=CHR$(8)
10005 PRINT"▒"
10010 OPENPG,PG
10015 PRINT#PG,CHR$(14)
10020 FORJ=0TO255STEP4
10025 PRINT#PG
10030 FORI=0TO3:PRINT#PG,CHR$(14);CHR$(1
6);I$(I);I+J;:NEXT:PRINT#PG
10040 FORK=0TO7:PRINT#PG,CHR$(14)"I";:FO
RI=0TO3
10050 X=PEEK(ZP+(J+I)*8+K)
10060 FORL=7TO0STEP-1:IFX>=2↑LTHENPRINT#
PG,"●";:X=X-2↑L:GOTO10080
10070 PRINT#PG," ";
10080 NEXT:PRINT#PG,"I";:NEXT:PRINT#PG,B
$:NEXT:NEXT
10085 PRINT#PG,CHR$(15)
10090 CLOSEPG
10100 RETURN

READY.
```



Disassembler für VC-20 & VC-64

In einer der nächsten Ausgaben wird in Homecomputer ein Assembler für den VC-20 abgedruckt werden. Heute zunächst ein Disassembler für diejenigen, die sich die Systemroutinen und aus Datastatements übernommene Programme in Maschinensprache näher betrachten möchten.

Wer in Maschinensprache programmiert - früher oder später beginnen die meisten Microcomputer-Freunde damit - der muß überprüfen können, was er programmiert hat.
Dazu verwendet man einen Disassembler. Der VC-20 hat ja in seiner Grundversion nur die Möglichkeit vorgesehen, mit PEEK, POKE, USR und SYS direkt auf seine Bits und Bytes zuzugreifen und an vorher bestimmte Speicherstellen zu springen um Maschinenprogramme aufzurufen.
Darum werden wir für die Homecomputerleser in den nächsten Ausgaben Serien von Routinen zum vereinfachten Programmieren in Assembler veröffentlichen.
Der hier vorgestellte Disassembler ist zwar in der Speicherausnutzung etwas verschwenderisch angelegt, ist aber dafür als Basicprogramm sehr schnell.
Beim VC-20 mit seiner Zeilenlänge von nur 22 Zeichen war es etwas schwierig, einen Weg zu finden, der das original 6502-Format richtig auf den Bildschirm bringt und trotzdem noch übersichtlich ist. Dank der farblichen Möglichkeiten des Gerätes ist dies aber recht gut zu lösen gewesen.
Nach dem Eintasten des Programms wird eine Dezimalzahl oder ein $-Zeichen, gefolgt von einer Hexzahl eingegeben. Danach folgt ein

```
D3F4-
20 E1    D3      JSR  $D3E1
D3F7-
A5 4F            LDA  $4F
D3F9-
48               PHA
D3FA-
A5 4E            LDA  $4E
D3FC-
48               PHA
D3FD-
20 F1    CE      JSR  $CEF1
D400-
20 8D    CD      JSR  $CD8D
D403-
68               PLA
D404-
85 4E            STA  $4E
D406-
68               PLA
D407-
85 4F            STA  $4F
```

Bildschirmausdruck VC-20

<L>, welches für List steht.
Nach Druck auf die Taste <RETURN> wird der VC-20 eine Bildschirmseite mit einem Assemlerlisting drukken.
Im folgenden genügt, sofern an der erreichten Stelle fortgefahren werden soll, die Eingabe <L> <RETURN>.
Das abgedruckte Disassembler-Programm wird Monat für Monat erweitert werden. Für diesen Zweck ist der Bereich zwischen den Zeilen 600 bis 799 freigehalten worden. Vorgesehen sind u.a. die Unterprogramme: Hex-Dump, ASCII, Commodore-Zeichen und Labeling.
Bezüglich des Programms für den VC-64 gilt das Gleiche, was zum VC-20 geschrieben wurde. Geändert werden mußte nur die Bildschirmausgabe, bei der aber trotz richtigem einzeiligem Format die Farben beibehalten wurden.

PGM für VC-20

```
10 POKE36879,25:PRINT"□";
100 DATABRK,ORA,???,???,???,ORA,ASL,???,
PHP,ORA,ASL,???,???,ORA,ASL,???
102 DATABPL,ORA,???,???,???,ORA,ASL,???,
CLC,ORA,???,???,???,ORA,ASL,???
104 DATAJSR,AND,???,???,BIT,AND,ROL,???,
PLP,AND,ROL,???,BIT,AND,ROL,???
106 DATABMI,AND,???,???,???,AND,ROL,???,
SEC,AND,???,???,???,AND,ROL,???
108 DATARTI,EOR,???,???,???,EOR,LSR,???,
PHA,EOR,LSR,???,JMP,EOR,LSR,???
110 DATABVC,EOR,???,???,???,EOR,LSR,???,
CLI,EOR,???,???,???,EOR,LSR,???
112 DATARTS,ADC,???,???,???,ADC,ROR,???,
PLA,ADC,ROR,???,JMP,ADC,ROR,???
114 DATABVS,ADC,???,???,???,ADC,ROR,???,
SEI,ADC,???,???,???,ADC,ROR,???
116 DATA???,STA,???,???,STY,STA,STX,???,
DEY,???,TXA,???,STY,STA,STX,???
118 DATABCC,STA,???,???,STY,STA,STX,???,
TYA,STA,TXS,???,???,STA,???,???
120 DATALDY,LDA,LDX,???,LDY,LDA,LDX,???,
TAY,LDA,TAX,???,LDY,LDA,LDX,???
122 DATABCS,LDA,???,???,LDY,LDA,LDX,???,
CLV,LDA,FSX,???,LDY,LDA,LDX,???
124 DATACPY,CMP,???,???,CPY,CMP,DEC,???,
INY,CMP,DEX,???,CPY,CMP,DEC,???
126 DATABNE,CMP,???,???,???,CMP,DEC,???,
CLD,CMP,???,???,???,CMP,DEC,???
128 DATACPX,SBC,???,???,CPX,SBC,INC,???,
INX,SBC,NOP,???,CPX,SBC,INC,???
130 DATABEQ,SBC,???,???,???,SBC,INC,???,
SED,SBC,???,???,???,SBC,INC,???
200 DATA1,2,1,1,1,2,2,1,1,2,1,1,1,3,3,1,
2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
202 DATA3,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1,
2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
204 DATA1,2,1,1,1,2,2,1,1,2,1,1,3,3,3,1,
```

Starship VC-20

```
2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
206 DATA1,2,1,1,1,2,2,1,1,2,1,1,3,3,3,1,
2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
208 DATA1,2,1,1,2,2,2,1,1,1,1,1,3,3,3,1,
2,2,1,1,2,2,2,1,1,3,1,1,1,3,1,1
210 DATA2,2,2,1,2,2,2,1,1,2,1,1,3,3,3,1,
2,2,1,1,2,2,2,1,1,3,1,1,3,3,3,1
212 DATA2,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1,
2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
214 DATA2,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1,
2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
300 DATA0,10,0,0,0,4,4,0,0,2,1,0,0,3,3,0
302 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
304 DATA3,10,0,0,4,4,4,0,0,2,1,0,3,3,3,0
306 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
308 DATA0,10,0,0,0,4,4,0,0,2,1,0,3,3,3,0
310 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
312 DATA0,10,0,0,0,4,4,0,0,2,1,0,12,3,3,
0
314 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
316 DATA0,10,0,0,4,4,4,0,0,0,0,0,3,3,3,0
318 DATA9,11,0,0,5,5,6,0,0,8,0,0,0,7,0,0
320 DATA2,10,2,0,4,4,4,0,0,2,0,0,3,3,3,0
322 DATA9,11,0,0,5,5,6,0,0,8,0,0,7,7,8,0
324 DATA2,10,0,0,4,4,4,0,0,2,0,0,3,3,3,0
326 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
328 DATA2,10,0,0,4,4,4,0,0,2,0,0,3,3,3,0
330 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
400 DATA,,,,#$,,$,,$,,$,",X",$,",Y",$,",
X",$,",Y",$,,($,",X)",($,"),Y",($,),
500 DIMM$(255),L%(255),F%(255),F1$(12),F
2$(12)
510 FORI=0TO255:READM$(I):NEXT:FORI=0TO2
55:READL%(I):NEXT:FORI=0TO255:READF%(I):
NEXT
515 FORI=0TO12:READF1$(I),F2$(I):NEXT
530 PC=0
600 INPUT"■";A$:IFA$="X"THENEND
602 PF=0:IFA$="LP"THENPF=1:GOTO1000
605 IFA$="L"THEN1000
610 IFRIGHT$(A$,1)="L"THENGOSUB800:GOTO1
000
690 GOTO600
800 IFLEFT$(A$,1)="$"THEN810
805 PC=VAL(A$):RETURN
810 PC=0:FORI=2TOLEN(A$)-1
820 A=ASC(MID$(A$,I,1))-48:IFA>9THENA=A-
7
830 PC=PC*16+A:NEXT:RETURN
900 FORJ=3TO0STEP-1:A=INT(DZ/16↑J):DZ=DZ
-A*16↑J:A=A+48:IFA>57THENA=A+7
910 PRINTCHR$(A);:NEXT:RETURN
950 FORJ=1TO0STEP-1:A=INT(DZ/16↑J):DZ=DZ
-A*16↑J:A=A+48:IFA>57THENA=A+7
960 PRINTCHR$(A);:NEXT:RETURN
1000 FORI=1TO11:DZ=PC:PRINT"◣";:GOSUB900
:PRINT"-▦":P1=PEEK(PC)
1005 ONL%(P1)GOSUB1100,1200,1300
1020 GOTO600
1100 DZ=P1:GOSUB950:PRINT"▦";TAB(10);M$(
P1):PC=PC+1:RETURN
1200 DZ=P1:GOSUB950:PRINT" ";:P2=PEEK(PC
+1):DZ=P2:GOSUB950
1205 PRINT"▦";TAB(10);M$(P1);
1210 IFF%(P1)=9THENP2=PEEK(PC+1):IFP2<12
8THENP2=PC+P2+2:GOTO1220
1215 IFF%(P1)=9THENP2=PC-(255-P2)+1
1220 PRINT" ";F1$(F%(P1));:DZ=P2:IFF%(P1
)=9THENGOSUB900:GOTO1240
1230 GOSUB950
1240 PRINTF2$(F%(P1)):PC=PC+2:RETURN
1300 DZ=P1:GOSUB950:PRINT" ";:P2=PEEK(PC
+1):DZ=P2:GOSUB950:PRINT" ";
1310 P3=PEEK(PC+2):DZ=P3:GOSUB950:PRINT"
▦";TAB(10);M$(P1);
1320 PRINT" ";F1$(F%(P1));:DZ=P3*256+P2:
```

```
E164-   2C A9 00    BIT   $00A9
E167-   85 0A       STA   $0A
E169-   20 D1 E1    JSR   $E1D1
E16C-   A5 0A       LDA   $0A
E16E-   A6 2B       LDX   $2B
E170-   A4 2C       LDY   $2C
E172-   20 D5 FF    JSR   $FFD5
E175-   B0 57       BCS   $E1CE
E177-   A5 0A       LDA   $0A
E179-   F0 1A       BEQ   $E195
E17B-   A2 1C       LDX   #$1C
E17D-   20 B7 FF    JSR   $FFB7
E180-   29 10       AND   #$10
E182-   F0 03       BEQ   $E187
E184-   4C 37 C4    JMP   $C437
E187-   A5 7A       LDA   $7A
E189-   C9 02       CMP   #$02
E18B-   F0 07       BEQ   $E194
E18D-   A9 64       LDA   #$64
E18F-   A0 C3       LDY   #$C3
E191-   4C 1E CB    JMP   $CB1E
E194-   60          RTS
E195-   20 B7 FF    JSR   $FFB7
```

Bildschirmausdruck VC-64

PGM für VC-64

Bis hier Zeilen vom VC-20-PGM übernehmen

```
900 FORJ=3TO0STEP-1:A=INT(DZ/16↑J):DZ=DZ
-A*16↑J:A=A+48:IFA>57THENA=A+7
910 PRINTCHR$(A);:NEXT:RETURN
950 FORJ=1TO0STEP-1:A=INT(DZ/16↑J):DZ=DZ
-A*16↑J:A=A+48:IFA>57THENA=A+7
960 PRINTCHR$(A);:NEXT:RETURN
1000 FORI=1TO24:DZ=PC:PRINT"▦";:GOSUB900
:PRINT"-    ▦";:P1=PEEK(PC)
1005 ONL%(P1)GOSUB1100,1200,1300
1010 NEXT:GOTO600
1100 DZ=P1:GOSUB950:PRINT"▦";TAB(20);M$(
P1):PC=PC+1:RETURN
1200 DZ=P1:GOSUB950:PRINT" ";:P2=PEEK(PC
+1):DZ=P2:GOSUB950
1205 PRINT"▦";TAB(20);M$(P1);
1210 IFF%(P1)=9THENP2=PEEK(PC+1):IFP2<12
8THENP2=PC+P2+2:GOTO1220
1215 IFF%(P1)=9THENP2=PC-(255-P2)+1
1220 PRINT"    ";F1$(F%(P1));:DZ=P2:IFF%(
P1)=9THENGOSUB900:GOTO1240
1230 GOSUB950
1240 PRINTF2$(F%(P1)):PC=PC+2:RETURN
1300 DZ=P1:GOSUB950:PRINT" ";:P2=PEEK(PC
+1):DZ=P2:GOSUB950:PRINT" ";
1310 P3=PEEK(PC+2):DZ=P3:GOSUB950:PRINT"
▦";TAB(20);M$(P1);
1320 PRINT"    ";F1$(F%(P1));:DZ=P3*256+P
2:GOSUB900:PRINTF2$(F%(P1)):PC=PC+3:RETU
RN

READY.
```

Starship VC-20

Slalom

Daß der Zeichensatz des VC-20 vom Anwender neu zu definieren ist, wurde in diesem Heft schon erwähnt. Hier ist nun ein Spiel, das einige selbst entworfene Zeichen benutzt.

Weil die Anzahl der Zeichen, die neu definiert wurden, nur fünf beträgt, lohnt es sich in diesem Falle nicht, den gesamten Zeichensatz vorzuladen, wie das zu tun wäre, wenn eine größere Menge neuer Zeichen vorhanden wäre.

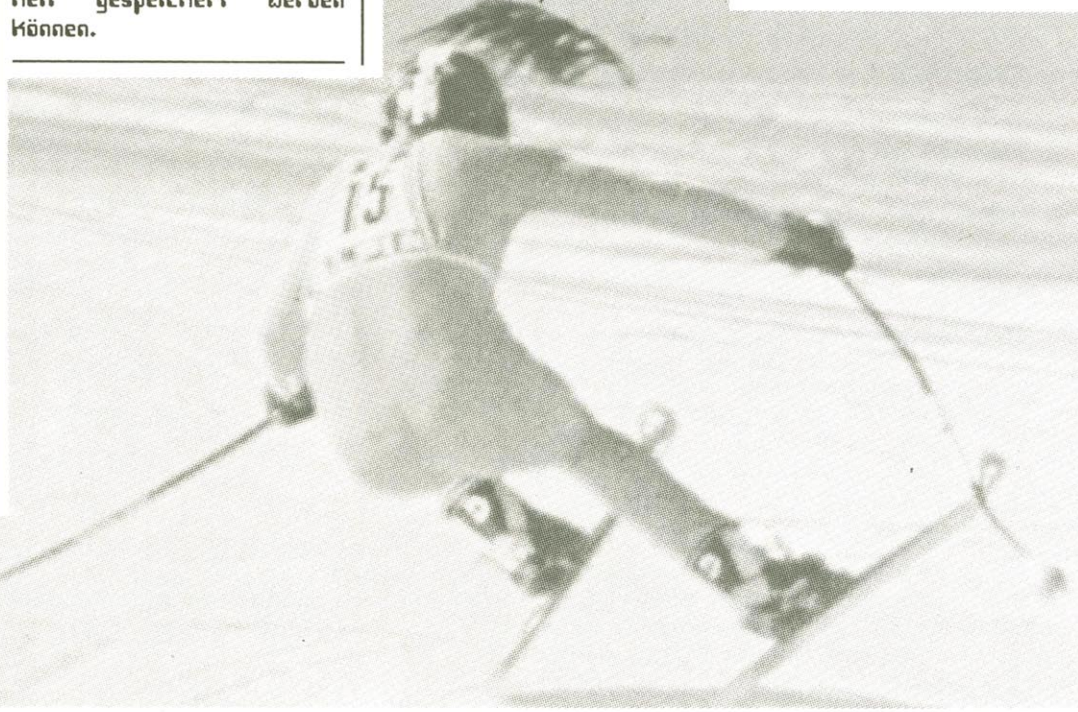
Es wurden deshalb die Zeilen 1 bis 8 eingefügt, die im einzelnen folgende Aufgaben erledigen:

1 29=Beginn des neuen Zeichensatzes : 02=Originalzeichensatz : Setze Zeiger auf neuen Zeichensatz

2 Übertrage Originalzeichensatz in den Bereich, den der neue Zeichensatz einnimmt.

3-7 Jeweils Zeichennummer des Zeichens, das verändert wird und 8 Codes mit den Informationen für das neue Zeichen

8 POKEd die neuen Zeichen in den Zeichensatz



Wichtig! POKE 44,28 POKE 7168,0

Vor jeden Laden des Programms und natürlich auch vor dem Eintasten muß mit POKE 44,28 : Poke 7168,0 der Basic-Startbereich verschoben werden, weil sonst der neue Zeichensatz das laufende Programm zerstört. Das Programm hat Highscore-Wertung und was besonders interessant ist, Tabellen, in denen die Mitspieler mit ihren erreichten Punktzahlen für die Ewigkeit gespeichert werden Können.

Tabellen steigern die Spannung

Nach RUN dauert es ein wenig, bis die Schleife beendet ist, die den neuen Zeichensatz herstellt. Danach erscheint auf dem Bildschirm ein Menü, mit einer Reihe Optionen. Falls schon Daten auf einer Kassette gespeichert sind, dann beginnt man mit LOAD DATEN.

Mit <E>, <T> und <P> Kann man sich die 3 verschiedenen Tabellen ansehen, die der VC-20 führt. In der "EWIGEN" befinden sich alle jemals erreichten Punktzahlen; in der TAGESBESTENLISTE erscheinen alle, an diesem Tage erreichten Ergebnisse und in der "PERSÖNLICHEN" sind alle Mitspieler, die jemals an einer Spielrunde teilnahmen mit ihrer besten Punktzahl vertreten.

Bevor mit <F> das Spielen beendet wird, werden die Daten mit <S> = SAVE DATEN auf eine separate Kassette überspielt - separat deshalb, weil dadurch vermieden wird, daß aus Versehen das Programm überschrieben wird.

Ein neues Spiel beginnt nach Drücken von <N>. Auf die Frage "Wer spielt?" benutzt man <+> und <->-Taste, um seinen eigenen Namen in der schon vorhandenen Namensliste zu finden. Ist der Name schon gespeichert, so drückt man <RETURN> und das Spiel beginnt. Befindet er sich noch nicht in der Liste, dann drückt man die <+>-Taste sooft, bis sich kein Name mehr zeigt und drückt dann <RETURN>. Der Name Kann jetzt eingegeben werden.

Bleibt nur noch zu erwähnen, daß man schon schnell reagieren muß, wenn man auf eine hohe Punktzahl Kommen will.

Starship VC-20

```
1 ZS=5120:OZ=32768:POKE36869,205
2 FORI=0TO2047:POKEI+ZS,PEEK(OZ+I):NEXT
3 DATA66,36,36,36,36,36,36,36,36
4 DATA76,64,96,112,80,80,64,64,224
5 DATA77,48,24,140,198,99,49,24,12
6 DATA78,12,24,49,99,198,140,24,48
7 DATA122,4,6,7,5,5,4,4,14
8 FORI=1TO5:READX:FORJ=0TO7:READY:POKEZS
+X*8+J,Y:NEXT:NEXT
10 DIMPP(60,2),II%(60,2),NL$(60),ST$(2):
R1=RND(-TI):HS=0
12 ST$(0)="┘    L":ST$(1)="┘    L":ST$(2)
="┘  L":S1=66:S2=77:S3=78
15 GOTO2000
20 PRINT"□":M=11:FR=0:PY=0
30 PZ=-5:F=-5:R1=-5:R2=-5:SL=S1
50 FORI=1TO17
110 POKE4162+M,SL:POKE37954+M,4
112 T=PEEK(197):SL=S1
114 IFT=33THENM=M-1:SL=S3:IFM<0THENM=0:S
L=S1
116 IFT=30THENM=M+1:SL=S2:IFM>21THENM=21
:SL=S1
120 PRINT:IFI<>1THEN130
122 IFM<=R1ORM>=R1+5-PYTHENPOKE36878,FR:
POKE36875,200:F=F+1
124 POKE36878,0:R1=R2
126 IFF=3THEN200
130 NEXT
165 PY=INT(PZ/10):PW=PY:IFPY>2THENPY=2
167 IFPY<0THENPY=0
170 R2=INT(RND(1)*16+1)
180 PRINTTAB(R2)ST$(PY)
183 IFPY=PZ/10THENPY=PY-1
184 IFPW=PZ/10THENPRINTPZ
185 PZ=PZ+1:IFPZ<>0THEN50
190 POKE36878,15:POKE36875,230:FORJ=1TO1
00:NEXTJ:POKE36878,0:F=0:FR=15:GOTO50
200 PZ=PZ-2
210 IFPZ<=HSTHEN250
220 POKE36875,0:POKE36878,15:FORI=1TO13:
FORJ=250TO240STEP-2:POKE36876,J:NEXTJ:
230 FORJ=240TO250:POKE36876,J:NEXTJ
240 POKE36876,0:NEXTI:POKE36878,0
250 PRINT"□PUNKTZAHL:";PZ
260 IFPZ>HSTHENHS=PZ
270 PRINT"HIGHSCORE:";HS
280 FORI=1TO100:NEXTI
1500 GOSUB50000
2000 PRINT"□"
2005 PRINT"■■■■";
2010 PRINT"L = LOAD DATEN"
2020 PRINT:PRINT"S = SAVE DATEN"
2040 PRINT:PRINT"E = EWIGE BESTENL."
2050 PRINT:PRINT"P = PERSOENL. BESTENL."
2055 PRINT"T = TAGESBESTENLISTE":PRINT
2060 PRINT"N = NEUES SPIEL":PRINT
2065 PRINT"F = ENDE":PRINT
2100 GETA$:IFA$="S"THEN3000
2110 IFA$="L"THEN4000
2130 IFA$="E"THENII=0:GOSUB6000:PRINT"□"
2140 IFA$="P"THENII=1:GOSUB6000:PRINT"□"
2150 IFA$="N"THEN8000
2155 IFA$="T"THENII=2:GOSUB6000:PRINT"□"
2160 IFA$="F"THENEND
2170 GOTO2005
3000 OPEN1,1,1,"TAB."
3010 FORI=1TO60
3012 IFNL$(I)=""THENNL$(I)="."
3015 PRINT#1,NL$(I):PRINT#1,STR$(PP(I,0)
):PRINT#1,STR$(PP(I,1))
3016 PRINT#1,STR$(II%(I,0)):PRINT#1,STR$
(II%(I,1)):NEXT
3020 CLOSE1
3030 GOTO2000
4000 OPEN1,1,0,"TAB."
4010 FORI=1TO60:INPUT#1,NL$(I),PE$,PP$,I
E$,IP$
4012 IFNL$(I)="."THENNL$(I)=""
4015 PP(I,0)=VAL(PE$):PP(I,1)=VAL(PP$):I
I%(I,0)=VAL(IE$):II%(I,1)=VAL(IP$):NEXT
4020 CLOSE1
4030 GOTO2000
6000 PRINT"■":LL=LEN(STR$(PP(1,II)))
6010 FORI=1TO60
6020 IFI/21=INT(I/21)THENWAIT197,32
6030 J=1:IFI>9THENJ=2
6040 PRINTMID$(STR$(I),J);
6050 PRINT".";LEFT$(NL$(II%(I,II)),17-LL
);TAB(20-LEN(STR$(PP(I,II))));PP(I,II)
6060 NEXT:WAIT197,32
6070 RETURN
8000 PRINT"□■WER SPIELT ?"
8005 I=1
8010 PRINT"■■■■"I;"■   ";NL$(I);"
"
8015 GETA$:IFA$=""THEN8015
8016 POKE650,128
8017 IFA$="+"THENI=I+1:IFI>60THENI=60
8020 IFA$="-"THENI=I-1:IFI<1THENI=1
8022 IFASC(A$)=13AND(NL$(I)=""ORI=60)THE
NINPUT"NAME";NL$(I)
8025 IFASC(A$)=13THENLS%=I:POKE650,0:GOT
O20
8030 GOTO8010
50000 GOSUB50100
50010 GOSUB50200
50015 GOSUB50100:II=2:GOSUB 50200
50020 GOSUB50100:II=1:TV=0
50030 GOSUB50400:RETURN
50100 TZ=PZ:TS%=LS%:FL=0:II=0:AP=60:RETU
RN
50200 FORI=1TOAP
50210 IFTZ=PP(I,II)ANDFL=1THENGOSUB50300
:GOTO50230
50220 IFTZ>PP(I,II)THENGOSUB50300
50230 NEXT
50240 RETURN
50300 FL=1:PP=PP(I,II):PP(I,II)=TZ:TZ=PP
50310 II%=II%(I,II):II%(I,II)=TS%:TS%=II
%:RETURN
50400 FORI=1TO60
50410 IFTS%=II%(I,II)ANDTZ>PP(I,II)THENP
P(I,II)=0:AP=I:GOSUB50200:RETURN
50420 IFTS%=II%(I,II)THENTV=1
50430 NEXT
50440 IFTV=0THENGOSUB50200
50450 RETURN
```

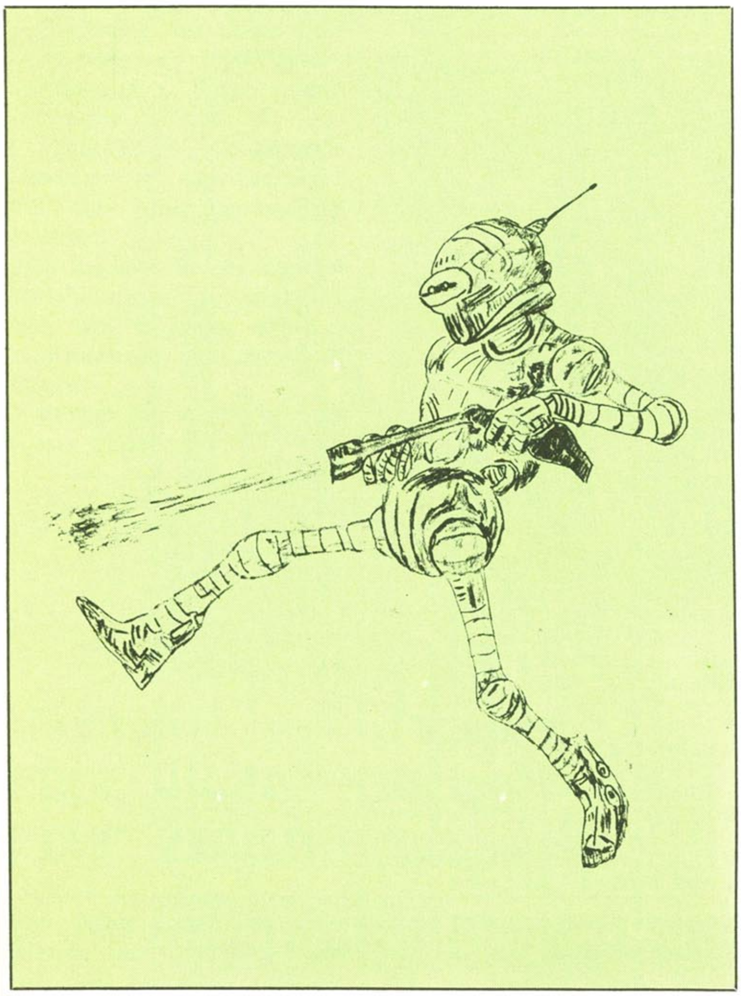
Robot

Mancher liebt Abschußspiele, mancher nicht. Bei Robot kommt es darauf an, eine Anzahl Roboter, die jedesmal an anderen Positionen auftauchen, zu treffen.

Zwei Striche bewegen sich nacheinander am unteren und linken Bildrand. Drückt man die Leertaste, so hält der erste Strich an und der zweite startet. Ausgangspunkt für beide Striche ist die Bildschirmecke unten links. Wird die Leertaste zum zweiten Mal gedrückt, dann wird an der Stelle, an der sich die verlängerten gedachten Linien der beiden Striche treffen, ein Trefferzeichen gezeichnet.

Trifft man genau die Mitte, so gilt der Robot als vollständig abgeschossen. Ansonsten hat man, sofern nicht alle aufgebraucht sind weitere Versuche.

Es werden Punkte vergeben, deren Höhe sich danach richtet, wie schnell der Robot abgeschossen wurde.



```
10 DIMST$(2),C%(6),C1%(2),D1(1),D2(1):R1
=RND(-TI):HS=0
20 GOSUB2000
40 GOSUB1100
50 GETA$:IFA$=""THEN50
55 PZ=0:PRINT""
60 FORM1=1TO5
100 B1=INT(RND(1)*16+3):B2=INT(RND(1)*16
)
110 B1=4118+B1+22*B2:B2=B1+33792
120 GOSUB1000
125 FORM2=1TO10
130 J1=0:J2=20:J3=1
140 C1%(0)=84:C1%(1)=66:C1%(2)=72
145 FORL=0TO1:K1=0:K2=2:K3=1
148 GETA$:IFA$="←"THEN600
150 GETA$:IFA$<>""THEN150
160 FORI=1TO2:FORJ=J1TOJ2STEPJ3
170 FORK=K1TOK2STEPK3:POKE4580+J,C1%(K):
POKE38372+J,0
180 GETA$:IFA$=" "THEND1(L)=J:D2(L)=K:GO
TO330
250 POKE4580+J,32:POKE38372+J,0
300 NEXTJ
310 J4=J1:J1=J2:J2=J4:J3=-J3:K1=2:K2=0:K
3=-1
320 NEXTI
325 D1(L)=J1:D2(L)=0
330 C1%(0)=82:C1%(1)=67:C1%(2)=69
340 J1=0:J2=-462:J3=-22
345 NEXTL
350 B3=4096+D1(0)+D1(1)+484:POKEB3,42:PO
KEB3+33792,2
355 PV=11-M2
360 IFB3=B1+1ANDC%(0)>42THENC%(0)=42:PZ=
PZ+11-M2:GOTO430
370 IFB3=B1+22ANDC%(1)>42THENC%(1)=42:PZ
=PZ+PV*.75:GOTO430
380 IFB3=B1+23THEN500
390 IFB3=B1+24ANDC%(3)>42THENC%(3)=42:PZ
=PZ+PV*.75:GOTO430
400 IFB3=B1+45ANDC%(4)>42THENC%(4)=42:PZ
=PZ+PV:GOTO430
410 IFB3=B1+66ANDC%(5)>42THENC%(5)=42:PZ
=PZ+PV*.5:GOTO430
420 IFB3=B1+68ANDC%(6)>42THENC%(6)=42:PZ
=PZ+PV*.5:GOTO430
425 GOTO440
430 POKE36878,15:POKE36877,200:FORI=1TO1
00:NEXTI:POKE36878,0:POKE36877,0
435 GOSUB1200:GOSUB1000
440 POKE4580+D1(0),32:POKE4580+D1(1),32
450 FORI=1TO300:NEXTI:GOTO580
500 POKE36877,220:FORL=15TO0STEP-1:POKE3
6878,L
510 IFL<7THENC%(L)=96:GOSUB1200:GOSUB100
0
520 FORI=1TO300:NEXTI:NEXTL
530 POKE36877,0:POKE36878,0
535 POKE4580+D1(0),96:POKE4580+D1(1),96
540 PZ=PZ+10*(11-M2)
570 GOTO585


580 PRINT"                            ":PRINT"
R=";M1;"S=";M2;"P=";PZ:NEXTM2
585 PRINT"                            ":PRINT"
R=";M1;"S=";M2;"P=";PZ
587 GOSUB1200:GOSUB1100
590 NEXTM1
600 IFPZ<=HSTHEN640
610 HS=PZ
620 POKE36878,15:FORM=1TO50:POKE36876,22
0
625 FORI=1TO15:NEXTI:POKE36876,0:NEXTM:F
ORM=1TO1000:NEXTM
630 POKE36878,0
640 PRINT"ERGEBNIS  :";PZ
660 PRINT"HIGHSCORE:";HS
670 PRINT"NEUES SPIEL:  LEERTASTE"
680 GOSUB1100
690 GOTO50
1000 POKEB1+1,C%(0):POKEB2+1,6
1010 POKEB1+22,C%(1):POKEB2+22,6
1020 POKEB1+23,C%(2):POKEB2+23,6
1030 POKEB1+24,C%(3):POKEB2+24,6
1040 POKEB1+45,C%(4):POKEB2+45,6
1050 POKEB1+66,C%(5):POKEB2+66,6
1060 POKEB1+68,C%(6):POKEB2+68,6
1070 RETURN
1100 C%(0)=81:C%(1)=80:C%(2)=224:C%(3)=7
9:C%(4)=224:C%(5)=255:C%(6)=127
1110 RETURN
1200 POKEB1+1,32:POKEB2+1,6
1210 POKEB1+22,96:POKEB2+22,6
1220 POKEB1+23,96:POKEB2+23,6
1230 POKEB1+24,96:POKEB2+24,6
1240 POKEB1+45,96:POKEB2+45,6
1250 POKEB1+66,96:POKEB2+66,6
1260 POKEB1+68,96:POKEB2+68,6
1270 RETURN
2000 PRINT"[Grafikzeichen]"
2010 PRINT"[Grafikzeichen]"
2020 PRINT"[Grafikzeichen]"
2030 PRINT"[Grafikzeichen]"
2040 PRINT"[Grafikzeichen]"
2050 PRINT"[Grafikzeichen]"
2060 PRINT"BY LUTZIAN WIDUCH 1982"
2065 FORI=1TO5000:NEXTI
2070 PRINT"SIE MUESSEN VERSUCHEN"
2080 PRINT"MIT MOEGLICHST WENIG"
2090 PRINT"SCHUESSEN DEN ROBOTER"
2100 PRINT"ABZUSCHIESSEN."
2102 PRINT "ZUM SCHIESSEN MUESSEN"
2103 PRINT "SIE DIE LEERTASTE"
2104 PRINT "DRUECKEN."
2110 PRINT"ALLES KLAR?"
2120 PRINT"DRUECKE LEERTASTE"
2140 RETURN
```


The Explorers Guide to the ZX
Mike Lord
Timedata
Programming Aid
Some Games, and
Other Novelties.
Applications.
Machine Language.
Discovering the ROM.
Hardware.
Edition 2
THE ACORN ATOM MAGIC BOOK
EXPLORING SPECTRUM BASIC
WAKE UP YOUR ATOM!
BRIAN LLOYD

Bundesliga



Computerfreunde mit Interesse am Sport haben viele Möglichkeiten, einen programmierbaren Rechner oder ein Microcomputersystem anzuwenden. Besonders in der Vereinsarbeit, aber auch bei Turnierveranstaltungen kann ein Computer wertvolle Hilfe leisten.

Er kann Spielpläne erstellen (das Programm folgt in einer der nächsten Ausgaben), Wettbewerbe protokollieren und Tabellen führen. Dem Letzteren wollen wir uns heute zuwenden.
Homecomputer stellt Ihnen ein Programm vor, mit dem vielseitige Möglichkeiten rund um die Handball- Volleyball- oder Fußball-Statistiken zu verwirklichen sind. Zum Beispiel die Simulation der folgenden Spieltage unter dem Motto "was wäre wenn...", oder das Erstellen von Sondertabellen die nur spezielle Fakten berücksichtigen.
Natürlich kann das Programm auch dazu herangezogen werden, eine laufende Spielrunde in Tabellenform festzuhalten und zu überwachen.
Eine besondere Eigenart des Programms ist, daß die Einzelgebnisse nicht unbedingt am Ende eines Spieltages eingegeben werden müssen, sondern auch schon im Tabellenbild berücksichtigt werden, während noch Partien im Gange sind. Dieser Service wurde für die Rundfunkhörer implementiert, die so neben einer laufenden Kontrolle über die Spielstände auch gleich die jeweils aktuelle Tabelle parat haben. Es dauert ja immer eine Weile, bis die Rundfunkleute nach Spielschluß die richtige Tabelle ermittelt haben.
Anhand der zahlreichen Beispielausdrucke ist es einfach, die einzelnen Möglichkeiten auszutesten.

```
]RUN

      L I G A - T A B E L L E N

 L = LADE TABELLE VON DISKETTE
 S = SAVE TABELLE AUF DISKETTE

 A = AENDERE SPIEL-ERGEBNISSE

 TH = DRUCKE HEIMTABELLE (BILDSCHIRM)
 TA = DRUCKE AUSWAERTSTABELLE (BILDSCH)
 TG = DRUCKE GESAMTTABELLE (BILDSCHIRM)
 #X = AUSGABE:SLOT 0 (X=0: BILDSCHIRM)

 K = KORRIGIERE TORE UND PUNKTE
 I = INITIALISIERE DIE TABELLEN
 W = WER GEGEN WEN?

 N = NAME DES TABELLENFILES ?
 C = CATALOG

 H = HOLE SPIELSTAND VON DISKETTE
 R = RETTE SPIELSTAND AUF DISKETTE
SLOT ? 1


 N = NAME DES TABELLENFILES ?
 C = CATALOG

 NAME ? FUSSBALL-BL
 LETZTES DATUM ? 28.3.1982
 HEUTIGES DATUM ? 30.3.1982


 K = KORRIGIERE TORE UND PUNKTE
 I = INITIALISIERE DIE TABELLEN
 W = WER GEGEN WEN?

HEIM,GAST ? 6,4
 2. HAMBURGER SV
 5. 1.FC KAISERSLTRN.
 7. VFL BOCHUM
 8. BORUSSIA DORTMUND
 9. VFB STUTTGART
10. KARLSRUHER SC
11. BAYER LEVERKUSEN
12. FORT. DUESSELDORF
14. MSV DUISBURG
16. DARMSTADT 98
17. WERDER BREMEN
18. EINTR. BRAUNSCHWG

1.FC NUERNBERG     -BAYERN MUENCHEN
1.FC KOELN         -ARMINIA BIELEFELD
EINTR. FRANKFURT   -BOR. M-GLADBACH


HEIM,GAST ? 18,17
 2. HAMBURGER SV
 5. 1.FC KAISERSLTRN.
 7. VFL BOCHUM
 8. BORUSSIA DORTMUND
 9. VFB STUTTGART
10. KARLSRUHER SC
11. BAYER LEVERKUSEN
12. FORT. DUESSELDORF
14. MSV DUISBURG
16. DARMSTADT 98

1.FC NUERNBERG     -BAYERN MUENCHEN
1.FC KOELN         -ARMINIA BIELEFELD
EINTR. FRANKFURT   -BOR. M-GLADBACH
EINTR. BRAUNSCHWG  -WERDER BREMEN


HEIM,GAST ? 5,16
 2. HAMBURGER SV
 7. VFL BOCHUM
 8. BORUSSIA DORTMUND
 9. VFB STUTTGART
10. KARLSRUHER SC
11. BAYER LEVERKUSEN
12. FORT. DUESSELDORF
14. MSV DUISBURG

1.FC NUERNBERG     -BAYERN MUENCHEN
1.FC KOELN         -ARMINIA BIELEFELD
EINTR. FRANKFURT   -BOR. M-GLADBACH
EINTR. BRAUNSCHWG  -WERDER BREMEN
1.FC KAISERSLTRN.  -DARMSTADT 98

HEIM,GAST ? 0,0
```

```
A = AENDERE SPIEL-ERGEBNISSE


 1.1.FC NUERNBERG -BAYERN MUENCHEN  0:0
 2.1.FC KOELN       -ARMINIA BIELEFE  0:0
 3.EINTR. FRANKFUR-BOR. M-GLADBACH  0:0
 4.EINTR. BRAUNSCH-WERDER BREMEN    0:0
 5.1.FC KAISERSLTR-DARMSTADT 98     0:0
REIHE,HEIMTORE,GASTTORE ? 1,0,3

 1.1.FC NUERNBERG -BAYERN MUENCHEN  0:3
 2.1.FC KOELN       -ARMINIA BIELEFE  0:0
 3.EINTR. FRANKFUR-BOR. M-GLADBACH  0:0
 4.EINTR. BRAUNSCH-WERDER BREMEN    0:0
 5.1.FC KAISERSLTR-DARMSTADT 98     0:0
REIHE,HEIMTORE,GASTTORE ? 2,0,1

 1.1.FC NUERNBERG -BAYERN MUENCHEN  0:3
 2.1.FC KOELN       -ARMINIA BIELEFE  0:1
 3.EINTR. FRANKFUR-BOR. M-GLADBACH  0:0
 4.EINTR. BRAUNSCH-WERDER BREMEN    0:0
 5.1.FC KAISERSLTR-DARMSTADT 98     0:0
REIHE,HEIMTORE,GASTTORE ? 3,3,0

 1.1.FC NUERNBERG -BAYERN MUENCHEN  0:3
 2.1.FC KOELN       -ARMINIA BIELEFE  0:1
 3.EINTR. FRANKFUR-BOR. M-GLADBACH  3:0
 4.EINTR. BRAUNSCH-WERDER BREMEN    0:0
 5.1.FC KAISERSLTR-DARMSTADT 98     0:0
REIHE,HEIMTORE,GASTTORE ? 4,1,1

 1.1.FC NUERNBERG -BAYERN MUENCHEN  0:3
 2.1.FC KOELN       -ARMINIA BIELEFE  0:1
 3.EINTR. FRANKFUR-BOR. M-GLADBACH  3:0
 4.EINTR. BRAUNSCH-WERDER BREMEN    1:1
 5.1.FC KAISERSLTR-DARMSTADT 98     0:0
REIHE,HEIMTORE,GASTTORE ? 5,3,1

 1.1.FC NUERNBERG -BAYERN MUENCHEN  0:3
 2.1.FC KOELN       -ARMINIA BIELEFE  0:1
 3.EINTR. FRANKFUR-BOR. M-GLADBACH  3:0
 4.EINTR. BRAUNSCH-WERDER BREMEN    1:1
 5.1.FC KAISERSLTR-DARMSTADT 98     3:1
REIHE,HEIMTORE,GASTTORE ? 0,0,0


TH = DRUCKE HEIMTABELLE (BILDSCHIRM)
TA = DRUCKE AUSWAERTSTABELLE (BILDSCH)
TG = DRUCKE GESAMTTABELLE (BILDSCHIRM)
#X = AUSGABE:SLOT 1 (X=0: BILDSCHIRM)

TABELLE FUSSBALL-BL

 1. BAYERN MUENCHEN     59:35     34:14
 2. 1.FC KOELN          48:21     34:16
 3. HAMBURGER SV        67:31     33:15
 4. BOR. M-GLADBACH     46:37     30:20
 5. WERDER BREMEN       39:34     29:17
 6. BORUSSIA DORTMUND   44:32     28:22
 7. VFB STUTTGART       43:38     26:22
 8. 1.FC KAISERSLTRN.   49:45     26:22
 9. EINTR. BRAUNSCHWG   48:44     26:22
10. VFL BOCHUM          38:36     25:27
11. EINTR. FRANKFURT    61:58     24:26
12. 1.FC NUERNBERG      39:54     22:28
13. KARLSRUHER SC       40:51     21:31
14. ARMINIA BIELEFELD   27:38     19:31
15. FORT. DUESSELDORF   39:59     19:33
16. BAYER LEVERKUSEN    30:54     17:31
17. MSV DUISBURG        33:57     17:35
18. DARMSTADT 98        32:58     16:34


HEIMTABELLE FUSSBALL-BL

 1. BAYERN MUENCHEN     44:13     24:2
 2. 1.FC KOELN          29:3      22:2
 3. EINTR. BRAUNSCHWG   34:13     22:2
 4. 1.FC KAISERSLTRN.   34:16     20:4
 5. HAMBURGER SV        43:12     19:3
 6. EINTR. FRANKFURT    39:20     19:5
 7. WERDER BREMEN       24:13     18:6
 8. VFL BOCHUM          23:13     18:6
 9. 1.FC NUERNBERG      24:18     18:6
10. KARLSRUHER SC       24:19     17:11
11. BOR. M-GLADBACH     28:17     16:8
12. FORT. DUESSELDORF   27:17     16:10
13. ARMINIA BIELEFELD   17:11     16:10
14. BORUSSIA DORTMUND   28:19     15:11
15. MSV DUISBURG        24:19     15:11
16. VFB STUTTGART       25:17     14:10
17. BAYER LEVERKUSEN    12:17     12:12
18. DARMSTADT 98        19:25     12:14
```

```
10 TA = 50
12 TB = 25: DIM T$(TA),TA(TA,2),TT(TA,5),TP(TA,5),VT(TA,5),VP(TA,5),WH(TB),WG(TB
),TH(TA),TG(TA),F(TA)
14  FOR I = 1 TO TA: FOR J = 0 TO 2:TA(I,J) = I: NEXT J,I
16 D$ =  CHR$ (4)
18  FOR I = 1 TO 40:S2$ = S2$ + " ": NEXT
20  IF DF = 1 THEN  RETURN
22  HOME : INVERSE : PRINT S2$: VTAB 2: PRINT "        L I G A - T A B E L L E N
     ": VTAB 3: PRINT S2$: NORMAL
24  PRINT " L = LADE TABELLE VON DISKETTE"
26  PRINT " S = SAVE TABELLE AUF DISKETTE"
28  PRINT : PRINT " A = AENDERE SPIEL-ERGEBNISSE"
30  PRINT : PRINT " TH = DRUCKE HEIMTABELLE (BILDSCHIRM)"
32  PRINT " TA = DRUCKE AUSWAERTSTABELLE (BILDSCH)"
34  PRINT " TG = DRUCKE GESAMTTABELLE (BILDSCHIRM)": PRINT " #X = AUSGABE:SLOT "
;D;" (X=0: BILDSCHIRM)"
35  PRINT : PRINT " K = KORRIGIERE TORE UND PUNKTE"
36  PRINT " I = INITIALISIERE DIE TABELLEN"
38  PRINT " W = WER GEGEN WEN?"
40  PRINT : PRINT " N = NAME DES TABELLENFILES ?": PRINT " C = CATALOG"
42  PRINT : PRINT " H = HOLE SPIELSTAND VON DISKETTE": PRINT " R = RETTE SPIELST
AND AUF DISKETTE"
44  VTAB 23: HTAB 39
46  GET A$: IF A$ = "L" THEN 700
48  IF A$ = "S" THEN 800
50  IF A$ = "A" THEN 300
52  IF A$ = "T" THEN 900
53  IF A$ = "K" THEN 2000
54  IF A$ = "R" THEN 500
56  IF A$ = "I" THEN 100
57  IF A$ = "#" THEN  HOME : PRINT "SLOT ? ";: GET A$:D =  VAL (A$): PRINT A$: G
OTO 22
58  IF A$ = "N" THEN  HOME : PRINT : INPUT "NAME ? ";NT$:NT$ = NT$ + " ": INPUT
"LETZTES DATUM ? ";LD$: INPUT "HEUTIGES DATUM ? ";HD$: GOTO 22
60  IF A$ = "C" THEN  HOME : GOTO 1000
62  IF A$ = "W" THEN 200
64  IF A$ = "H" THEN 600
66  GOTO 44
100  HOME
102 F = 1
104  IF F > WT THEN WT = F
106  HOME : FOR I = 1 TO WT
108  IF I < 10 THEN  PRINT " ";
110  PRINT I"   .................";: HTAB 7: PRINT T$(I);: HTAB 3: IF F = I THE
N  INPUT ". ? ";T$: IF  LEN (T$) > 17 THEN 118
112  IF T$ >  < "" THEN T$(I) = T$
113  IF T$ = "O" THEN WT = WT - 1
114 T$ = "": PRINT
116  NEXT : GOTO 120
118  PRINT "NAME ZU LANG": CALL  - 198: GOTO 110
120  INPUT "NUMMER ? ";F: IF F > 1 THEN  GOSUB 104: GOTO 120
122  HOME : INPUT "TORE EINGEBEN ? ";JN$: IF  LEFT$ (JN$,1) = "N" THEN 128
124  FOR I = 1 TO WT: PRINT T$(I): PRINT "HEIM"; TAB( 12);: INPUT "H,A ? ";VT(I,
0),VT(I,1): PRINT "AUSWAERTS"; TAB( 12);: INPUT "H,A ? ";VT(I,2),VT(I,3):VT(I,4)
 = VT(I,0) + VT(I,2):VT(I,5) = VT(I,1) + VT(I,3)
126  PRINT "GESAMT ="; TAB( 12);VT(I,4)":"VT(I,5): PRINT : NEXT
128  HOME : INPUT "PUNKTE EINGEBEN ? ";JN$: IF  LEFT$ (JN$,1) = "N" THEN 134
130  FOR I = 1 TO WT: PRINT T$(I): PRINT "HEIM"; TAB( 12);: INPUT "H,A ? ";VP(I,
0),VP(I,1): PRINT "AUSWAERTS"; TAB( 12);: INPUT "H,A ? ";VP(I,2),VP(I,3):VP(I,4)
 = VP(I,0) + VP(I,2):VP(I,5) = VP(I,1) + VP(I,3)
132  PRINT "GESAMT ="; TAB( 12);VP(I,4)":"VP(I,5): PRINT : NEXT
134  GOSUB 1200: GOSUB 350: GOTO 22
200 WG =  INT (WT / 2)
202  FOR I = 1 TO WT:F(I) = 0:TH(I) = 0:TG(I) = 0: NEXT
204  FOR I = 1 TO WG:WH(I) = 0:WG(I) = 0: NEXT
206  HOME : FOR I = 1 TO WT: IF F(I) > 0 THEN 214
208  IF I < 10 THEN  PRINT " ";
210  PRINT I". ";T$(I)
212  IF I / 20 =  INT (I / 20) THEN  GET A$
214  NEXT :I = 0
216  PRINT
218 I = I + 1: IF I <  = WG THEN  IF WH(I) >  < 0 THEN  PRINT T$(WH(I)): TAB( 19
);"-";T$(WG(I)): GOTO 218
220  PRINT
222  IF I <  = WG THEN  INPUT "HEIM,GAST ? ";WH(I),WG(I):F(WH(I)) = 1:F(WG(I)) =
 1: IF WH(I) >  < 0 THEN 206
224 WG = I - 1: GOTO 22
300  PR# D: PRINT : HOME : FOR I = 1 TO WG: IF I < 25 THEN  VTAB I
302  IF I < 10 THEN  PRINT " ";
304  PRINT I;".";
306  PRINT  LEFT$ (T$(WH(I)),15);: HTAB 19: PRINT "-"; LEFT$ (T$(WG(I)),15);: HT
AB 35: PRINT " ";
308  IF TH(WH(I)) < 10 THEN  PRINT " ";
310  PRINT TH(WH(I))":"TG(WG(I))
312  NEXT
314  PR# 0: INPUT "REIHE,HEIMTORE,GASTTORE ? ";I,TH,TG: PR# D: PRINT "REIHE,HEIM
TORE,GASTTORE ? ";I","TH","TG: IF I =  < WG THEN TH(WH(I)) = TH:TG(WG(I)) = TG:
IF I = 0 THEN  PR# 0: GOTO 318
316  GOTO 300
318  FOR I = 1 TO WG
320 TT(WH(I),0) = VT(WH(I),0) + TH(WH(I))
322 TT(WH(I),1) = VT(WH(I),1) + TG(WG(I))
324 TT(WG(I),2) = VT(WG(I),2) + TG(WG(I))
326 TT(WG(I),3) = VT(WG(I),3) + TH(WH(I))
328 TT(WH(I),4) = TT(WH(I),0) + TT(WH(I),2)
330 TT(WH(I),5) = TT(WH(I),1) + TT(WH(I),3)
332 TT(WG(I),4) = TT(WG(I),0) + TT(WG(I),2)
334 TT(WG(I),5) = TT(WG(I),1) + TT(WG(I),3)
336  NEXT
338  FOR I = 1 TO WG
340  IF TH(WH(I)) = TG(WG(I)) THEN TP(WH(I),0) = VP(WH(I),0) + 1:TP(WH(I),1) = V
P(WH(I),1) + 1:TP(WG(I),2) = VP(WG(I),2) + 1:TP(WG(I),3) = VP(WG(I),3) + 1
342  IF TH(WH(I)) < TG(WG(I)) THEN TP(WH(I),1) = VP(WH(I),1) + 2:TP(WG(I),2) = V
P(WG(I),2) + 2:TP(WH(I),0) = VP(WH(I),0):TP(WG(I),3) = VP(WG(I),3)
344  IF TH(WH(I)) > TG(WG(I)) THEN TP(WH(I),0) = VP(WH(I),0) + 2:TP(WG(I),3) = V
P(WG(I),3) + 2:TP(WH(I),1) = VP(WH(I),1):TP(WG(I),2) = VP(WG(I),2)
346 TP(WH(I),4) = TP(WH(I),0) + TP(WH(I),2):TP(WH(I),5) = TP(WH(I),1) + TP(WH(I)
,3):TP(WG(I),4) = TP(WG(I),0) + TP(WG(I),2):TP(WG(I),5) = TP(WG(I),1) + TP(WG(I)
,3)
348  NEXT : GOSUB 350: GOTO 22
350 X = 0
352 Y = X * 2:F0 = 2:F1 = WT
354 FL = 0: VTAB 23: PRINT X,F0" ",F1" ": FOR I = F0 TO F1: IF TP(TA(I - 1,X),Y)
 > TP(TA(I,X),Y) THEN 378
356  IF TP(TA(I - 1,X),Y) = TP(TA(I,X),Y) THEN 362
358 TQ = TA(I - 1,X):TA(I - 1,X) = TA(I,X):TA(I,X) = TQ:FL = 1:F1 = I
360  GOTO 378
362  IF TP(TA(I - 1,X),Y + 1) < TP(TA(I,X),Y + 1) THEN 378
364  IF TP(TA(I - 1,X),Y + 1) = TP(TA(I,X),Y + 1) THEN 368
366  GOTO 358
368  IF TT(TA(I - 1,X),Y) - TT(TA(I - 1,X),Y + 1) > TT(TA(I,X),Y) - TT(TA(I,X),Y
```

```
 + 1) THEN 378
370  IF TT(TA(I - 1,X),Y) - TT(TA(I - 1,X),Y + 1) = TT(TA(I,X),Y) - TT(TA(I,X),Y
 + 1) THEN 374
372  GOTO 358
374  IF TT(TA(I - 1,X),Y) >  = TT(TA(I,X),Y) THEN 378
376  GOTO 358
378  NEXT
380  IF FL = 0 THEN 408
382 FL = 0: VTAB 23: PRINT X,FO" ",F1" ": FOR I = F1 TO FO STEP  - 1: IF TP(TA(I
 - 1,X),Y) > TP(TA(I,X),Y) THEN 406
384  IF TP(TA(I - 1,X),Y) = TP(TA(I,X),Y) THEN 390
386 TQ = TA(I - 1,X):TA(I - 1,X) = TA(I,X):TA(I,X) = TQ:FL = 1:FO = I
388  GOTO 406
390  IF TP(TA(I - 1,X),Y + 1) < TP(TA(I,X),Y + 1) THEN 406
392  IF TP(TA(I - 1,X),Y + 1) = TP(TA(I,X),Y + 1) THEN 396
394  GOTO 386
396  IF TT(TA(I - 1,X),Y) - TT(TA(I - 1,X),Y + 1) > TT(TA(I,X),Y) - TT(TA(I,X),Y
 + 1) THEN 406
398  IF TT(TA(I - 1,X),Y) - TT(TA(I - 1,X),Y + 1) = TT(TA(I,X),Y) - TT(TA(I,X),Y
 + 1) THEN 402
400  GOTO 386
402  IF TT(TA(I - 1,X),Y) >  = TT(TA(I,X),Y) THEN 406
404  GOTO 386
406  NEXT : IF FL > 0 THEN 354
408 X = X + 1: IF X < 3 THEN 352
410  RETURN
500  PRINT
505  PRINT D$;"OPEN SP.ST.";NT$;HD$
510  PRINT D$;"DELETESP.ST."NT$;HD$
515  PRINT D$;"OPEN SP.ST.";NT$;HD$
520  PRINT D$;"WRITESP.ST.";NT$;HD$
525 I = 1
530  PRINT T$(I): PRINT F(I): PRINT TH(I): PRINT TG(I)
532 I = I + 1: IF T$(I) >  < "" THEN 530
533  PRINT "XXXXX"
535  FOR I = 1 TO WG: PRINT T$(WH(I)): PRINT T$(WG(I)): NEXT
537  PRINT "XXXXX"
540  PRINT D$;"CLOSESP.ST."NT$;HD$
545  GOTO 22
600  PRINT
605  PRINT D$;"OPEN SP.ST.";NT$;HD$
610  PRINT D$;"READ SP.ST.";NT$;HD$
617  INPUT T$: IF T$ = "XXXXX" THEN 623
618  FOR I = 1 TO 50: IF T$ = T$(I) THEN 620
619  NEXT :I = 0: PRINT "ERROR"
620  INPUT F(I): INPUT TH(I): INPUT TG(I): GOTO 617
623 I = 1
625  INPUT T$: IF T$ = "XXXXX" THEN 633
626  FOR J = 1 TO 50: IF T$ = T$(J) THEN WH(I) = J: GOTO 628
627  NEXT
628  INPUT T$: FOR J = 1 TO 50: IF T$ = T$(J) THEN WG(I) = J: GOTO 630
629  NEXT
630 I = I + 1:WG = I - 1: GOTO 625
633  PRINT D$;"CLOSESP.ST."NT$;HD$
635  GOTO 22
700  PRINT
702  PRINT D$;"OPEN TAB."NT$;LD$
704  PRINT D$;"READ TAB."NT$;LD$
706  INPUT WT
708  FOR I = 1 TO WT
710  INPUT T$(I)
712  FOR J = 0 TO 2: INPUT TA(I,J): NEXT
714  FOR J = 0 TO 5: INPUT VT(I,J): INPUT VP(I,J):TT(I,J) = VT(I,J):TP(I,J) = VP
(I,J): NEXT
716  NEXT
718  PRINT D$;"CLOSETAB."NT$;LD$
720  GOTO 22
800  PRINT
802  PRINT D$;"OPEN TAB."NT$;HD$
804  PRINT D$;"DELETETAB."NT$;HD$
806  PRINT D$;"OPEN TAB."NT$;HD$
808  PRINT D$;"WRITETAB."NT$;HD$
810  PRINT WT
812  FOR I = 1 TO WT
814  PRINT T$(I)
816  FOR J = 0 TO 2: PRINT TA(I,J): NEXT
818  FOR J = 0 TO 5: PRINT TT(I,J): PRINT TP(I,J): NEXT
820  NEXT
822  PRINT D$;"CLOSETAB."NT$;HD$
824  GOTO 22
900  GET A$: HOME : PR# D
901  IF A$ = "A" THEN Y = 2: PRINT "AUSWAERTSTABELLE ";
902  IF A$ = "H" THEN Y = 0: PRINT "HEIMTABELLE ";
904  IF A$ = "G" THEN Y = 4: PRINT "TABELLE ";
906 X = Y / 2
908  PRINT NT$: PRINT :CV = 2
909  IF RF = 1 THEN 911
910  FOR I = 1 TO WT: VTAB 24:CV = CV + 1: IF CV < 24 THEN  VTAB CV
911  IF I < 10 THEN  PRINT " ";
912  PRINT I;". ";T$(TA(I,X));: HTAB 24
914  IF TT(TA(I,X),Y) < 100 THEN  PRINT " ";
916  IF TT(TA(I,X),Y) < 10 THEN  PRINT " ";
918  PRINT TT(TA(I,X),Y);":";TT(TA(I,X),Y + 1);: HTAB 34
920  IF TP(TA(I,X),Y) < 100 THEN  PRINT " ";
922  IF TP(TA(I,X),Y) < 10 THEN  PRINT " ";
924  PRINT TP(TA(I,X),Y);":";TP(TA(I,X),Y + 1)
925  IF RF = 1 THEN  RETURN
926  NEXT
928  GET A$: IF A$ = " " THEN  PRINT : PRINT : PR# 0: GOTO 22
930  GOTO 928
1000  PRINT
1002  PRINT D$;"CATALOG"
1004  GET A$: GOTO 22
1200  FOR I = 1 TO WT: FOR J = 0 TO 5:TT(I,J) = VT(I,J):TP(I,J) = VP(I,J): NEXT
J,I: RETURN
2000  HOME : FOR J = 0 TO 2 STEP 2:A$ = "H": IF J = 2 THEN A$ = "A"
2002  FOR I = 1 TO WT:RF = 1: HOME : GOSUB 901
2004 RF = 0: PRINT : PRINT "KORREKTUR (J/N) ": GET A$: IF A$ = "N" THEN 2008
2006  INPUT "+TORE ? ";TT(TA(I,X),Y): INPUT "-TORE ? ";TT(TA(I,X),Y + 1): INPUT
"+PKTE ? ";TP(TA(I,X),Y): INPUT "-PKTE ? ";TP(TA(I,X),Y + 1)
2008  NEXT
2010  NEXT
2012  FOR I = 1 TO WT
2014 TT(I,4) = TT(I,0) + TT(I,2):TT(I,5) = TT(I,1) + TT(I,3):TP(I,4) = TP(I,0) +
 TP(I,2):TP(I,5) = TP(I,1) + TP(I,3)
2016  NEXT
2018 A$ = ""
2020  GOTO 318
```

DIVE-BOMBER

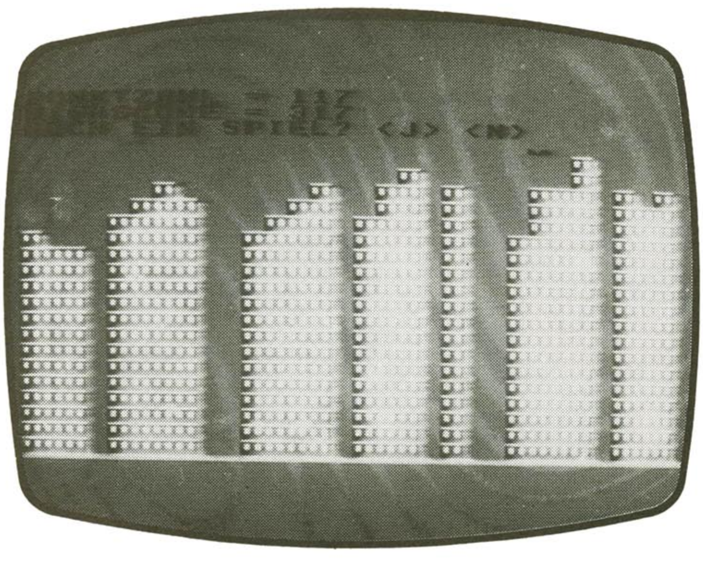
Ein Spielprogramm für den TI-99/4 in TI-BASIC.

Bei Dive-Bomber soll ein Flugzeug so lange wie möglich über eine Reihe Hochhäuser fliegen, ohne diesen zu nahe zu kommen. Da das Flugzeug laufend tiefer sinkt, kann dies nur durch gezielten Abwurf von Bomben verhindert werden, wodurch die Häuser Stockwerk um Stockwerk eliminiert werden.

Da der TI-99 in siner Grundversion keine vernünftigen Druckbefehle kennt, wurde eine Routine angehängt, die, wenn auch äußerst langsam, Texte an eine beliebige Stelle auf den Bildschirm schreibt.

Das Programm beginnt mit dem Initialisieren der RND-Funktion und dem Neu-Definieren einiger Zeichen und Farben. Danach werden die Häuser per Zufallszahlengenerator aufgebaut und das Flugzeug startet am oberen linken Bildrand. Durch Drücken einer Taste können Bomben abgeworfen werden, die die Stockwerke vom Einschlagpunkt aus nach oben löschen.

Wird vom Programm erkannt, daß das Flugzeug ein Haus streifen wird, ohne daß dies vom Spieler verhindert werden kann, so wird das Spiel abgebrochen und akustisch angezeigt, ob die alte Bestmarke (Highscore) übertroffen wurde.



```
100 RANDOMIZE
110 CALL CLEAR
120 CALL CHAR(96,"00FF999999
FFFFFF")
130 CALL CHAR(112,"000000000
086CFFE")
140 CALL CHAR(104,"FFFFFF")
150 CALL CHAR(42,"0000006078
06")
160 CALL COLOR(9,5,16)
170 CALL COLOR(11,2,1)
180 CALL COLOR(10,15,1)
190 REM START NEUES SPIEL
200 PZ=0
210 REM HAEUSER AUFBAUEN
220 CALL CLEAR
230 I=2
240 J=RND*8+3
250 FOR B=1 TO RND*5
260 CALL VCHAR(J,I,96,24-J)
270 I=I+1
280 IF I>31 THEN 320
290 NEXT B
300 I=I+1
310 IF I<32 THEN 240
320 CALL HCHAR(24,1,104,32)
330 REM START BOMBER
340 FX=1
350 FY=1
360 LX=0
370 LY=1
380 LX=LX+1
390 IF LX<32 THEN 420
400 LY=LY+1
410 LX=1
420 CALL SOUND(-1000,-6,24-F
Y)
430 FX=FX+1
440 IF FX<32 THEN 470
450 FX=1
460 FY=FY+1
470 CALL GCHAR(FY,FX,T)
480 IF T=96 THEN 800
490 CALL GCHAR(FY+1,FX+1,T)
500 IF T=96 THEN 510 ELSE 52
0
510 IF BZ<2 THEN 800
520 CALL HCHAR(FY,FX,112)
530 CALL HCHAR(LY,LX,32)
540 PZ=PZ+1
550 CALL KEY(0,KEY,S)
560 IF S=0 THEN 610
570 IF BZ>0 THEN 620
580 BX=FY
590 BX=FX
600 BZ=1
610 IF BZ=0 THEN 730
620 CALL HCHAR(BY,BX,32)
630 BX=BX+1
640 BY=BY+1
650 IF BX<32 THEN 680
660 BX=1
670 BY=BY+1
680 IF BY>22 THEN 760
690 CALL GCHAR(BY,BX,X)
700 IF X=96 THEN 760
710 CALL HCHAR(BY,BX,42)
720 BZ=BZ+1
730 IF FY<23 THEN 380
740 IF FX<31 THEN 380
750 GOTO 210
760 CALL VCHAR(1,BX,32,BY)
770 CALL SOUND(-2000,-5,0)
780 BZ=0
790 GOTO 730
800 DRUCK$="PUNKTZAHL = "&ST
R$(PZ)
810 VTAB=1
820 HTAB=2
830 GOSUB 960
840 IF HS>PZ THEN 870
850 HS=PZ
860 GOSUB 1140
870 DRUCK$="HIGHSCORE= "&ST
R$(HS)
880 GOSUB 960
890 DRUCK$="NOCH EIN SPIEL?
<J> <N> "
900 GOSUB 960
910 CALL KEY(0,KEY,STATUS)
920 IF STATUS=0 THEN 910
930 IF KEY=74 THEN 190
940 IF KEY<>78 THEN 910
950 END
960 REM PRINTROUTINE
970 FOR DRI=1 TO LEN(DRUCK$)
980 CODE=ASC(SEG$(DRUCK$,DRI
,1))
990 GOSUB 1030
1000 NEXT DRI
1010 GOSUB 1080
1020 RETURN
1030 REM COUT
1040 CALL HCHAR(VTAB,HTAB,CO
DE)
1050 HTAB=HTAB+1
1060 IF HTAB<32 THEN 1130
1070 REM CROUT
1080 VTAB=VTAB+1
1090 HTAB=2
1100 IF VTAB<25 THEN 1130
1110 VTAB=24
1120 PRINT
1130 RETURN
1140 FOR SI=1 TO 5
1150 FOR SJ=-3 TO -1
1160 CALL SOUND(100,SJ,1)
1170 NEXT SJ
1180 NEXT SI
1190 RETURN
```

250 Computer zu gewinnen





Eines dieser Geräte kann schon bald in Ihrem Hause stehen. Nur ein bischen Werbung für Homecomputer im Kollegen- und Freundes-Kreis und Sie können Ihren Traum-Computer gewinnen!

Daß Homecomputer anders ist als andere Computermagazine haben Sie schon bemerkt. Allein in diesem Heft sind 20 Programme für die populärsten Microcomputer abgedruckt.

Darum haben wir uns auch etwas besonderes einfallen lassen, als es darum ging, eine Abonnenten-Werbeaktion zu starten.
Je mehr Leser Homecomputer regelmäßig beziehen, um so besser können wir das Magazin für Sie gestalten.

Denn Abonnenten bedeuten in unserem jetzigen Stadium mehr Mitarbeiter. Und mehr Mitarbeiter, das bedeutet, daß jeder mehr Sorgfalt in seine Programmierarbeit legen kann. Die Qualität steigt und das kommt Ihnen, dem Leser zugute.

Jeder, der einen Abonnenten für Homecomputer wirbt, erhält als kleines Dankeschön ein Taschenbuch zum Thema Microcomputer. Außerdem verteilen wir 250 Microcomputer nach Wahl an diejenigen Leser, die Homecomputer die meisten Leser zuführen.

Die Gewinnverteilung erfolgt nach einem, für alle Leser interessanten Schlüssel:
Jedesmal, wenn bei uns 1000 neue Abonnenten registriert wurden, dann erhalten diejenigen 10 Leser, die zu diesem Zeitpunkt die meisten Abonnenten geworben haben - also in der Wettbewerbsliste vorne liegen, einen Microcomputer Ihrer Wahl; und zwar der Erste dieser Wettbewerbsliste einen Microcomputer im Werte von 2500 DM und die 9 Folgenden jeder einen im Werte von 1000,-DM.
Die 10 Gewinner werden aus der Wettbewerbsliste gestrichen (sie können allerdings wieder neu beginnen und nochmal gewinnen), und die ihnen Folgenden rücken nach.
Sind wieder 1000 neue Abonnenten registriert, dann geht es wie oben beschrieben weiter.
Teilnahmeberechtigt ist jeder, der Homecomputer regelmäßig bezieht.
Der Rechtsweg ist ausgeschlossen.
Die Gewinner werden sofort benachrichtigt und in der jeweils folgenden Ausgabe von Homecomputer bekanntgegeben.

Bestellkarte

Ich möchte Homecomputer ab sofort zum günstigen Abonnementpreis von 55,– DM, für 12 Ausgaben, monatlich ins Haus geliefert bekommen.

Name/Vorname

Straße PLZ Ort

Ich wünsche folgende Zahlungsweise (12 Hefte jährlich DM 55,– innerhalb der BRD, Ausland s. Impressum)

☐ **Bargeldlos und bequem durch Bankeinzug:** ______ BLZ (vom Scheck abschreiben)

Konto-Nr. ______ Geldinstitut ______

☐ **Gegen Rechnung** (keine Vorauszahlung leisten)

Datum/Unterschrift

Diese Karte ausschneiden oder Fotokopieren und einsenden an umseitige Adresse.

Ich habe einen Abonnementen für Homecomputer geworben

Seine Adresse:

Ich beziehe zu den mir bekannten Bedingungen ab sofort regelmäßig Homecomputer zum Abonnemntpreis von 55,– DM.

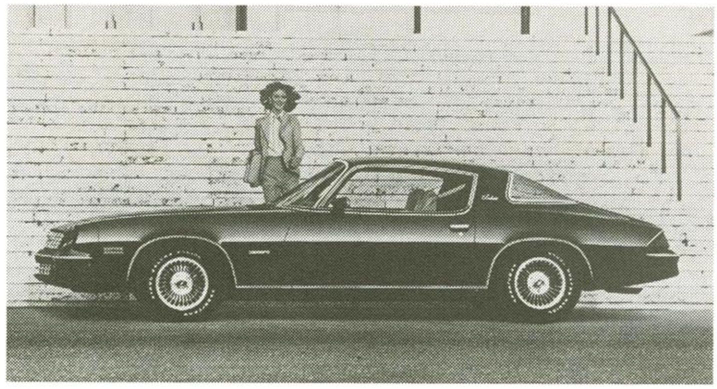
Datum Unterschrift des neuen Abonnenten

Senden Sie das Microcomputertaschenbuch an meine Adresse und registrieren Sie mich in der Wettbewerbsliste:

Adresse des Werbers: ______

Diese Karte ausschneiden oder Fotokopieren und einsenden an umseitige Adresse.

AutoKosten



Jeder, der ein Auto fährt, stellt sich früher oder später die Frage: Was kostet mich das Fahrzeug?

Hat man einen Computer zur Verfügung, dann kann man sich die Kosten ohne langen Zeitaufwand nach Kriterien geordnet ausrechnen lassen und die Kosten pro Kilometer errechnen.

Das abgedruckte Programm erfüllt alle diese Wünsche und noch einiges mehr.

Garantie

Wir senden Ihnen Homecomputer regelmäßig ab der nächsterreichbaren Ausgabe zu.

★

Die Lieferung erfolgt frei Haus inclusive Mehrwertsteuer und Zustellgebühren.

★

Sie können Ihr Homecomputer-Abonnement jeweils am Jahresende kündigen (Mindestbezugsdauer 1 Jahr). Die Kündigungsfrist beträgt 8 Wochen.

Bitte frei machen

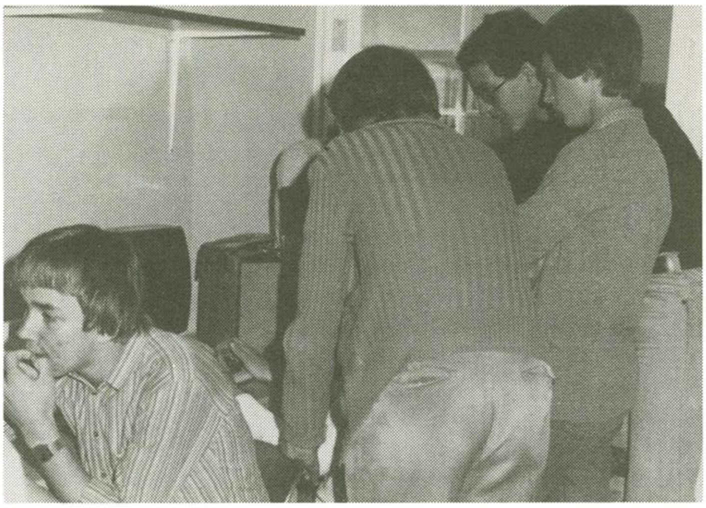
Homecomputer
Leserservice
Gartenstraße 47
3440 Eschwege

```
10 D$ =  CHR$ (4)
20 L$ = "                                      "
90  GOTO 1000
100  HOME : INVERSE : FOR I = 1 TO 39: PRINT " ";: NEXT : PRINT
110  PRINT " "; TAB( 15);"AUTOKOSTEN"; TAB( 39);" "
120  FOR I = 1 TO 39: PRINT " ";: NEXT : PRINT
130  NORMAL : PRINT : RETURN
150  INVERSE : VTAB 4
155  PRINT " :POSTEN      :KM     :DATUM    :GES.KOSTN"
160  NORMAL : PRINT : RETURN
200  DATA 9
201  DATA "0 FAHRZEUG"
202  DATA "1 KRAFTSTOFF"
205  DATA "2 OEL"
210  DATA "3 STEUER"
215  DATA "4 VERSICHERNG."
220  DATA "5 REIFEN"
225  DATA "6 REPARATUREN"
230  DATA "7 PFLEGE"
235  DATA "8 LUXUSZUBEH."
240  DATA "9 HEUTE"
300  READ AD: FOR I = 0 TO AD: READ K$(I): NEXT
310  RETURN
350  IF ZW < .01 THEN ZW$ = "      0.00": RETURN
355 ZW = ZW + .00501:ZW$ =  STR$ (ZW):L =  LEN ( STR$ ( INT (ZW)))
360 ZW$ =  LEFT$ (ZW$,L + 3)
370  IF  LEN (ZW$) < 9 THEN ZW$ = " " + ZW$: GOTO 370
380  RETURN
400  FOR I = 0 TO AD: IF I = 9 THEN  PRINT
401  PRINT K$(I); LEFT$ (L$,39 -  LEN (K$(I))): NEXT : GOSUB 500
402  VTAB 6: FOR I = 0 TO AD: IF I = 9 THEN  PRINT
403  HTAB 15:ZW$ = "         " +  STR$ (LK(I)): PRINT  RIGHT$ (ZW$,6): NEXT
404  VTAB 6: FOR I = 0 TO AD: IF I = 9 THEN  PRINT
405  HTAB 22:ZW$ = "00" +  STR$ (LT(I)): PRINT  RIGHT$ (ZW$,2);".";:ZW$ = "00" +  STR$ (LM(I)):
PRINT  RIGHT$ (ZW$,2);".";:ZW$ = "00" +  STR$ (LJ(I)): PRINT  RIGHT$ (ZW$,2): NEXT
406  VTAB 6: FOR I = 0 TO AD: IF I = 9 THEN  PRINT
407  HTAB 31:ZW = GP(I): GOSUB 350: PRINT ZW$: NEXT
410  GOSUB 600
415  PRINT : PRINT "<L> = LADE / <S> = SAVE DATEN AUF DISK"
420  PRINT "<0> - <";AD;"> = EINGABE NEUER KOSTEN"
430  INPUT "NACH EINGABE <RETURN> DRUECKEN ? ";E$
440  IF E$ = "L" THEN  GOTO 800
445  IF E$ = "S" THEN  GOTO 900
450  IF  VAL (E$) < 0 OR  VAL (E$) > AD THEN  RETURN
460  GOSUB 100:E =  VAL (E$): ON E GOTO 2100,2200,2300,2400,2500,2600,2700,2800,2900
465  GOTO 2000
500 GP(9) = 0: FOR I = 0 TO 8:GP(9) = GP(9) + GP(I): NEXT
502  GOSUB 700
505  FOR I = 0 TO 8: IF LK(9) < LK(I) THEN LK(9) = LK(I)
506  NEXT
510 KM = LK(9) - LK(0): IF KM = 0 THEN 590
512 ZW = GM / (LK(1) - LK(0)) * 100: GOSUB 350:SH$ = ZW$
515  IF  LEFT$ (SH$,1) = " " THEN SH$ =  MID$ (SH$,2): GOTO 515
520 ZW = GP(9) / KM * 100: GOSUB 350:KO$ = ZW$
521  IF  LEFT$ (KO$,1) = " " THEN KO$ =  MID$ (KO$,2): GOTO 521
530 ZW = 0: FOR I = 1 TO BA:ZW = ZW + BM(I): NEXT :ZK = BK(BA) - BK(0): IF ZK > 0 THEN ZW = ZW /
 ZK * 100: GOSUB 350:BK$ = ZW$
535  IF  LEFT$ (BK$,1) = " " THEN BK$ =  MID$ (BK$,2): GOTO 535
540 TG = (LJ(9) - LJ(0)) * 365 + (LM(9) - LM(0)) * 30 + (LT(9) - LT(0)):TM = TG / 30:TJ = TG / 3
65
545 ZW = GP(9) / TM: GOSUB 350:TM$ = ZW$:ZW = GP(9) / TJ: GOSUB 350:TJ$ = ZW$
546  IF  LEFT$ (TM$,1) = " " THEN TM$ =  MID$ (TM$,2): GOTO 546
547  IF  LEFT$ (TJ$,1) = " " THEN TJ$ =  MID$ (TJ$,2): GOTO 547
590  RETURN
600  PRINT : PRINT "VERBR.: LTR/100KM ="SH$;"  Z.ZT.=";BK$
610  PRINT "KOSTEN: PFG/KM =";KO$
615  IF TM > 1 THEN  PRINT "             DM/MON =";TM$;
620  IF TJ > 1 THEN  PRINT " DM/JR =";TJ$;
625  PRINT
690  RETURN
700 RT = LJ(9) * 365 + LM(9) * 31 + LT(9)
705  FOR I = 0 TO 8: IF LJ(I) * 365 + LM(I) * 31 + LT(I) > RT THEN LJ(9) = LJ(I):LM(9) = LM(I):L
T(9) = LT(I): GOTO 700
710  NEXT
715  RETURN
800  GOSUB 100: INPUT "KFZ-KENNZEICHEN ? ";KF$
810  PRINT D$;"OPENAUTO: ";KF$
820  PRINT D$;"READAUTO: ";KF$
850  FOR I = 0 TO 9: INPUT LT(I),LM(I),LJ(I),LK(I),GP(I): NEXT
855  FOR I = 0 TO 5: INPUT BK(I),BM(I): NEXT
860  INPUT BA,GM,GO
870  PRINT D$"CLOSEAUTO: ";KF$
880  RETURN
900  GOSUB 100: INPUT "KFZ-KENNZEICHEN ? ";KF$
910  PRINT D$;"OPENAUTO: ";KF$
920  PRINT D$;"DELETEAUTO: ";KF$
930  PRINT D$;"OPENAUTO: ";KF$
940  PRINT D$;"WRITEAUTO: ";KF$
950  FOR I = 0 TO 9: PRINT LT(I): PRINT LM(I): PRINT LJ(I): PRINT LK(I): PRINT GP(I): NEXT
955  FOR I = 0 TO 5: PRINT BK(I): PRINT BM(I): NEXT
960  PRINT BA: PRINT GM: PRINT GO
970  PRINT D$"CLOSEAUTO: ";KF$
980  RETURN
1000  GOSUB 300
1010  GOSUB 100
1015  GOSUB 150
1020  GOSUB 400
1030  GOTO 1010
2000  IF LK(0) > 0 THEN 2050
2010  INPUT "ANSCHAFFUNGSDATUM (T,M,J)! ? ";LT(0),LM(0),LJ(0)
2015  IF LJ(0) < 1900 THEN LJ(0) = LJ(0) + 1900
2020  PRINT : INPUT "ANSCHAFFUNGSPREIS DM...... ? ";GP(0)
2030  PRINT : INPUT "BEI KILOMETERSTAND ....... ? ";LK(0):BK(0) = LK(0)
2040  RETURN
2050  INPUT "ZAHLUNG ? ";RZ:GP(0) = GP(0) + RZ: RETURN
2100  INPUT "LITER ? ";BM: IF BM = 0 THEN  RETURN
2105  PRINT : INPUT "PREIS ? ";BP
2107  GOSUB 3000
2110 BA = BA + 1: IF BA = 6 THEN  FOR I = 1 TO 5:BK(I - 1) = BK(I):BM(I - 1) = BM(I): NEXT :BA =
 BA - 1
2115 BM(BA) = BM:GP(1) = GP(1) + BP:GM = GM + BM:BK(BA) = BK
2120  RETURN
2200  INPUT "LITER ? ";OM: IF OM = 0 THEN  RETURN
2205  PRINT : INPUT "PREIS ? ";OP
2207  GOSUB 3000
2210 GP(2) = GP(2) + OP:GO = GO + OM
2220  RETURN
2300  INPUT "STEUER (DM) ? ";GP:GP(3) = GP(3) + GP
2310  GOSUB 3000
2320  RETURN
2400  INPUT "VERSICHERUNG (DM) ? ";GP:GP(4) = GP(4) + GP
2410  GOSUB 3000
2420  RETURN
2500  INPUT "REIFEN (DM) ? ";GP:GP(5) = GP(5) + GP
2510  GOSUB 3000
2520  RETURN
2600  INPUT "REPARATURKOSTEN ? ";GP:GP(6) = GP(6) + GP
2610  GOSUB 3000
2620  RETURN
2700  INPUT "PFLEGEKOSTEN ? ";GP:GP(7) = GP(7) + GP
2710  GOSUB 3000
2720  RETURN
2800  INPUT "LUXUSZUBEHOER (DM) ? ";GP:GP(8) = GP(8) + GP
2810  GOSUB 3000
2820  RETURN
3000  PRINT : INPUT "DATUM (T,M,J) ? ";LT(E),LM(E),LJ(E)
3005  IF LJ(E) < 1900 THEN LJ(E) = LJ(E) + 1900
3010  PRINT : INPUT "BEI KM-STAND ? ";LK(E):BK = LK(E)
3020  RETURN
```

4000 Augen = 1 A.U.G.E.

A.U.G.E.

Wenn zwei Deutsche die gleichen Interessen haben, dann gründen sie einen Verein. Am Beispiel einer Regionalgruppe der Apple User Group Europe informierte sich Homecomputer, was die Mitglieder des Computervereins erleben.



Immer, wenn der zweite Samstag im Monat naht, klingelt bei den Herren Heil, Schnaut und Zersch das Telefon öfter als gewöhnlich. Die Club-Mitglieder aus den umliegenden Bezirken fragen beim Regionalleiter, Stellvertreter oder Clubtreffen-Oranisator an, ob das nächste Club-Treffen wie geplant am bekannten Ort, ein Gymnasium in Warburg, stattfindet.

Kurz vor zwei Uhr füllt sich das Klassenzimmer und die ersten Gespäche beleben den Raum, Joysticks wechseln ihre Besitzer, die schuleigenen Apples werden vorbreitet.

Kurz nach Zwei ist es dann so weit. 20 bis 30 junge Männer - nicht ein weibliches Wesen zu sehen - tragen sich in die Anwesenheitsliste ein und lauschen mehr oder weniger begeistert den Ausführungen des Regionalleiters Heil.

Dieser verliest die Clubmitteilungen "von oben", die vom Vorstand an alle Regionalgruppen versandt werden und macht Anmerkungen zu den einzelnen Punkten.

Dabei entwickeln sich leicht erste Diskussionen, die gelegentlich recht engagiert geführt werden.

Ein besonderes Thema der letzten Treffen war eine hitzige schriftliche Auseinandersetzung zwischen Vertretern der Fa. Apple und einzelnen Clubmitgliedern, die sich von der Firma vernachlässigt behandelt fühlten. Wie immer in einem solchen Fall, wenn sich die Fronten verhärten, kommt es zu Kontroversen Ansichten über das weitere Verhalten.

Der augenblickliche Stand der Dinge ist wohl, daß alle wieder in die gleiche Richtung rudern wollen; sitzen doch alle im gleichen Boot.

Nach etwa einer Stunde, so lange kann die Erörterung der Clubthemen dauern, geht man dann schließlich zum gemütlichen Teil über. Das bedeutet, die einzelnen Mitglieder bilden Grüppchen, in denen sich die einzelnen Interessen wiederspiegeln. Die Spiele-Enthusiasten zeigen neu entwickelte Spiele, die Theoretiker diskutieren sich die Köpfe über Lösungsmöglichkeiten von Algorithmen heiß und die Software-Sammler kopieren die Vereins-Software für ihre Sammlung.

Und schließlich ist es Zeit zum Aufbruch und man hätte so gerne noch dieses Thema angesprochen und jenes Programm vorgeführt.

Aber das nächste Treffen kommt bestimmt und so fährt man nach Hause um einige Informationen reicher, gestärkt vom Gefühl mit Gleichgesinnten seinem Hobby gefröhnt zu haben und setzt sich an seinen Apple, vom Gedanken beseelt, daß das Club-Treffen eigentlich noch etwas länger hätte dauern können.



BRITAIN'S LARGEST SINGLE MICRO USER GROUP

BEEBUG FOR THE BBC MICRO

INDEPENDENT NATIONAL USER GROUP FOR THE BBC MICRO

MEMBERSHIP NOW EXCEEDS 10,000

BEEBUG runs a regular magazine devoted exclusively to the BBC Micro (10 issues per year). Now 36 pages.
First issue April 1982. Reprints of all issues available to members.
Programs — Hints and Tips — Major articles — News — Reviews — Commentary.
PLUS members' discount scheme PLUS members' software library.

April Issue: 3D Noughts and Crosses, Moon Lander, Ellipse and 3D Surface. Plus articles on Upgrading to Model B, Making Sounds, and Operating System Calls.
May Issue: Careers, Bomber, Chords, Spiral and more.
Plus articles on Graphics, Writing Games Programs, and Using the Assembler.
June Issue: Mazetrap, Mini Word Processor, Polygon; plus articles on Upgrading, The User Port, TV set and Monitor Review, Graphics Part II, More Assembler Hints, Structuring in BBC Basic, plus BBC Bugs.
July Issue (36 pages) Invaders and patchwork programs. Fix for BBC cassette Bugs, Mode 7 explored, User define Keys, Software reviews, High res graphics printout, RS423 receive fix.
September Issue: Games programs! Higher/Lower, Hangman, plus string search program. Articles on debugging programs, a Safe Verifier, new User guide, errata, creating and moving multicolour characters, logic and the Beeb, ideas on colouring and shading, Acorn's ROM replacement charge plus much more.
October Issue. Program features: alien attack, calendar generator, Union Jack, memory display utility. Articles on debugging, improving key detection, Acorn press release on O.S.1.2. issue II Basic, the tube and second processor options, a new series for less experienced users, software reviews.

Membership
6 months £4.90, 1 year £8.90
Send £1.00 and A4 SAE for sample
Overseas 1 year only:
Europe £15.00, Middle East £18.00
Americas & Africa £20.00,
Other countries £22.00.

Make cheques to BEEBUG and send to BEEBUG, Dept 4, 374 Wandsworth Rd., London, SW8 4TE.

---

campbell systems

THE VERY BEST IN MACHINE CODE FOR THE **ZXSPECTRUM** & **ZX81**

SPECTRUM 16K **GULPMAN** game of the '.....man' variety with 15 mazes, 4 chasers, laser defence, 9 grades, 9 speeds, demo mode, choice of joystick control. "An extraordinarily good program" raves Boris Allan for Popular Computing Weekly. We think you will agree. £5.95

SPECTRUM 48K **MASTERFILE** business/domestic filing and reporting system. So flexible that it is equally usable for your mailing lists, catalogues, stock control, text extracts ...... applications are endless. Fully user-defined data and report display formats, dynamic variable-length file, records and data items. Fully menu-driven with powerful search facilities, sorting, total/average, update, multiple independent files, printing. Yes, we aim to support microdrive when Uncle delivers. Nearly all the 8K we use is machine code, so you get 32K per file. Comes with example file and 22-page manual. £15.00

SPECTRUM 16K **SPDE** Disassembler and Editor, as used by other ZX professionals, and we used it to develop the above. £5.95

ZX81 16-48K **THE FAST ONE** is the predecessor to MASTERFILE and is in use all over the world now. Specification is very similar to MASTERFILE. £12.00

ZX81 16K **GULP 2** almost identical spec to GULPMAN. £4.75

All programs supplied double-recorded and mailed 1st class by return. Prices include VAT and postage within Europe. SAE for full list.

CAMPBELL SYSTEMS
(Dept.HC)
15 ROUS ROAD
BUCKHURST HILL
ESSEX IG9 6BL
ENGLAND 01-504-0589

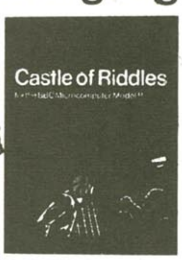
CS

WANTED!

PROFESSIONAL ADVENTURERS TO UNDERTAKE THREE DANGEROUS MISSIONS

You will need a quick agile mind, a rock-steady hand, and nerves of steel to survive. Your reward will be treasure beyond the dreams of man and the satisfaction of going where no man has gone before.

Special Competition with over £3,000 in prizes.

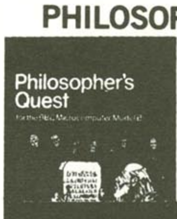


CASTLE OF RIDDLES

Win over £3,000 in prizes in the fabulous Acornsoft/Your Computer competition. Entry form on every pack.

Enter the warlock's castle, solve the riddles, and regain the ring to become 'King of the Ring'.

PHILOSOPHER'S QUEST



A fascinating adventure in a world of fiendish puzzles to be solved. To complete your quest—think hard about everything you do.

SPHINX ADVENTURE



A full-sized classic adventure game in which you move through caves, fight with trolls, collect treasure and finally make your way to the sphinx to collect your reward.

All the adventures can be run on the BBC Microcomputer Model B and cost £9.95 including post and packing.

Acornsoft Limited c/o Vector Marketing Ltd, Denington Industrial Estate, Wellingborough, Northants NN8 2RL.

To: Acornsoft Limited c/o Vector Marketing Ltd, Denington Industrial Estate Wellingborough, Northants NN8 2RL.
Please send me the following adventure games.

Philosophers Quest	SBG01 . . . (qty) at £9.95 =	
Sphinx Adventure	SBG07 . . . (qty) at £9.95 =	
Castle of Riddles	SBG17 . . . (qty) at £9.95 =	
		Total ______

I enclose cheque no ______________ or please debit my Access or Barclaycard no

Please Print
Name: ______
Address: ______
Postcode: ______
Signature

End of File 3/83

Im nächsten Heft zum Lesen und zum Abtippen unter anderem:

VC-64 musiziert in HiFi

Labyrinthe - in 2-D / 3-D und mit Monstern, Mäusen und Geistern

Apple-Spiele mit superschneller Sprite-Grafik und natürlich das Sprite-Grafikprogramm zur eigenen Spieleherstellung

VC-20 Assembler und Space-Invaders in Maschinensprache

Survive für ZX-81

Reportage über die Herstellung von Homecomputer

Und viele weitere Programme und Überraschungen

Leser-Software-Service

Nicht jeder Leser hat die Zeit und die Geduld, alle Programme, die er gerne hätte, in sein Gerät einzutasten.
Wir haben uns deshalb überlegt, daß es das einfachste ist, wenn wir die Programme, die wir ohnehin auf Diskette oder Kassette vorliegen haben, diesem Leserkreis zum Selbstkostenpreis (Datenträger, Kopierzeitaufwand, Porto, Verpackung und Mehrwertsteuer) anbieten.
Gegen Einsendung eines Schecks oder Vorauszahlung auf unser Konto bei der Kreissparkasse Eschwege, Nummer 45 22 934 senden wir Ihnen die gewünschten Programme schnellstmöglich zu.
Es genügt jeweils die Angabe eines Stichwortes, z.B.: VC-20-Disk Heft 3 oder Sinclair-Kassette Heft 3.
K = Kassette
D = Diskette

VC-64 K 8,-DM
...Biorhythmus
...Roulette
...Labyrinth
...Disassembler

CBM (Pet 3000) K 8,-DM
...Concorde
...Uhr
...Kalender
...51
...Roulette

Sharp MZ 80 K 5,-DM
...Submarine-Hunt

ZX-81 K 8,-DM
...U-Boot-Jagd
...Rotamint

Apple II D 15,-DM
...Ascroll/KBD
...Diskhandler
...Bundesliga
...Autokosten

VC-20 K 8,-DM
. D 15,-DM
...Zeichengenerator
...Disassembler
...Slalom
...Robot

TI 99/4 K 5,-DM

Auftrag für Gelegenheitsanzeigen in Homecomputer

An Homecomputer
Gartenstraße 47
3440 Eschwege

Name und Adresse ______________________

______________________ Abo.-Nr. ________

Unterschrift   Ich zahle sofort nach Rechnungserhalt.   Datum

Ich wünsche folgenden Text zu veröffentlichen:

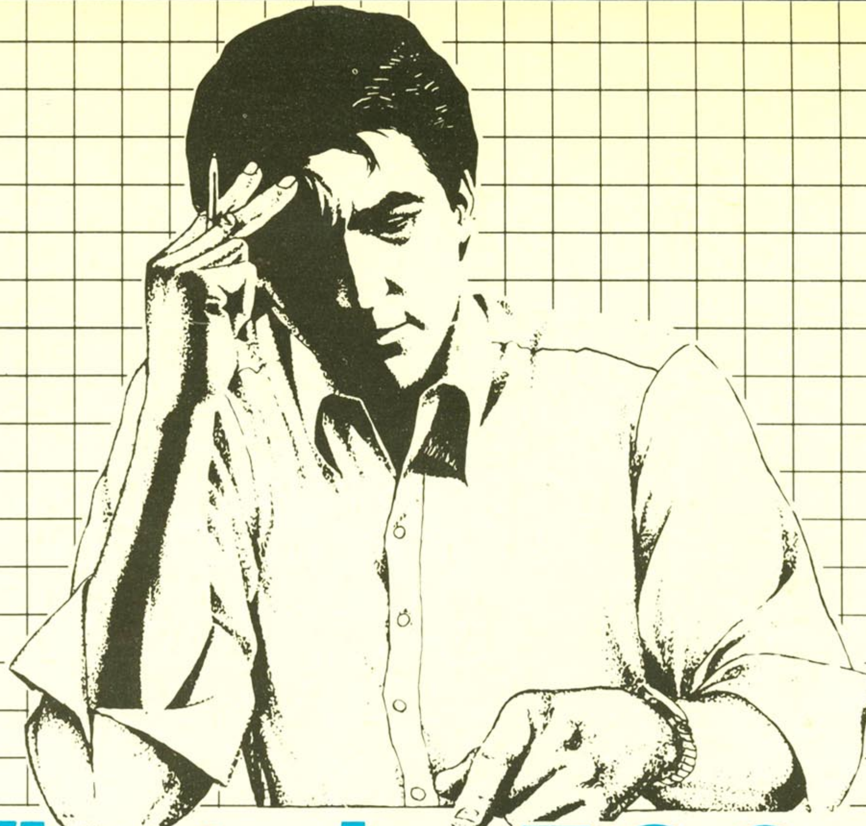
Unter der Rubrik „Kleinanzeigen" veröffentlichen wir Gelegenheitsanzeigen für Verkaufsangebote, Kauf- und Tauschgesuche, Kontaktaufnahme bzw. Erfahrungsaustausch usw.

Preise für „Kleinanzeigen": Private Gelegenheitsanzeige je Druckzeile 5,— DM inkl. MwSt. Chiffregebühr je Anzeige 10,– DM.

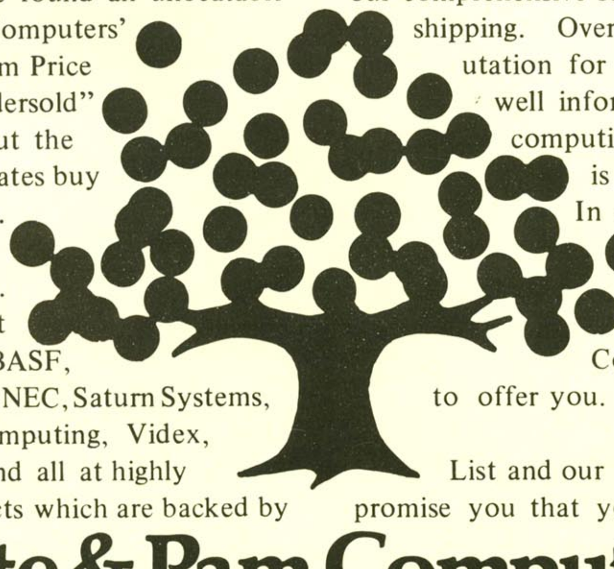
Gewerbliche Gelegenheitsanzeige je Druckzeile 7,— DM inkl. MwSt. (dürfen nicht unter Chiffre erscheinen).

Zutreffendes ankreuzen

suche ☐ Software   suche ☐ Hardware   ☐ Tausch   ☐ Versch.
biete an ☐ Software   biete an ☐ Hardware   ☐ Kontakte   ☐ Chiffre



What do 500 computer dealers know that you don't?



They know that they have found an unbeatable combination – Pete & Pam Computers' service, with the Pete & Pam Price Promise, "We will not be undersold" Over 500 dealers throughout the UK, Europe and the Gulf States buy from Pete & Pam Computers. They choose from over 900 products in our catalogue. Products from 146 different companies such as Apple, BASF, Epson, Microsoft MicroPro, NEC, Saturn Systems, Sirius Software, Versa Computing, Videx, VisiCorp and Zenith – and all at highly competitive prices. Products which are backed by our comprehensive service facilities, with same-day shipping. Over 500 dealers know our reputation for sound advice, and for being well informed in what's new in microcomputing. They know our service is friendly, fast and efficient. In short – they know us. Isn't it time you were "in the know" and opened an account with Pete & Pam Computers? We have a lot to offer you.

Send for our free Price List and our product literature. We promise you that you will not be disappointed.

Pete & Pam Computers

Head Office: New Hall Hey Road, Rossendale, Lancashire, BB4 6JG Tel 0706 227011 Telex 635740 PETPAM G

London Office: 103-5 Blegborough Road, London, SW16 6DL Tel 01-769 1022/3/4 & 01-677 2052 Telex 923070 PPCOMP G